

UNIVERZA V LJUBLJANI
FAKULTETA ZA RAČUNALNIŠTVO IN INFORMATIKO

Gregor Šekoranja

**Izdelava sistema za upravljanje
spletnih vsebin z uporabo ogrodja
MODx**

DIPLOMSKO DELO
UNIVERZITETNI ŠTUDIJSKI PROGRAM PRVE STOPNJE
RAČUNALNIŠTVO IN INFORMATIKA

MENTOR: izr. prof. dr. Marjan Krisper

Ljubljana, 2014

Rezultati diplomskega dela so intelektualna lastnina avtorja. Za objavljanje ali izkoriščanje rezultatov diplomskega dela je potrebno pisno soglasje avtorja, Fakultete za računalništvo in informatiko ter mentorja.

Besedilo je oblikovano z urejevalnikom besedil \LaTeX .

Fakulteta za računalništvo in informatiko izdaja naslednjo nalogo:

Tematika naloge:

V okviru diplomske naloge razvijte sistem, ki bo omogočal enostavno in hitro urejanje dinamičnih in odzivno oblikovanih spletnih vsebin preko uporabniškega in zalednega vmesnika. Sistem naj omogoča razvoj poljubnih zalednih komponent in popolno svobodo programerju pri izbiri tehnologij za dosego poljubnih ciljev uporabnika brez večjih omejitev. Poleg urejanja vsebin naj bo za sistemske administratorje omogočeno tudi urejanje programske kode kar preko brskalnika. Posebno pozornost pri razvoju sistema posvetite tudi varnosti, predvsem avtentikaciji in pravicam uporabnikov. Zaželeno je tudi, da ima sistem vgrajeno komponento za upravljanje z dodanimi komponentami. Možna naj bo tudi združitev več spletnih domen v enem samem sistemu.

IZJAVA O AVTORSTVU DIPLOMSKEGA DELA

Spodaj podpisani Gregor Šekoranja, z vpisno številko **63010141**, sem avtor diplomskega dela z naslovom:

Izdelava sistema za upravljanje spletnih vsebin z uporabo ogrodja MODx.

S svojim podpisom zagotavljam, da:

- sem diplomsko delo izdelal samostojno pod mentorstvom izr. prof. dr. Marjana Krisperja;
- so elektronska oblika diplomskega dela, naslov (slov., angl.), povzetek (slov., angl.) ter ključne besede (slov., angl.) identični s tiskano obliko diplomskega dela;
- soglašam z javno objavo elektronske oblike diplomskega dela na svetovnem spletu preko univerzitetnega spletnega arhiva.

V Ljubljani, dne 15. septembra 2014

Podpis avtorja:

Zahvaljujem se mentroju izr. prof. dr. Marjanu Krisperju za pomoč in vodenje pri opravljanju diplomske naloge. Posebna zahvala velja ženi, družini in staršem, ki so me vseskozi spodbujali in mi omogočili študij.

Diplomsko nalogo posvečam svoji predragi družini, saj je bila največji motiv, da uspešno zaključim dodiplomski študij.

Kazalo

Povzetek

Abstract

1	Uvod	1
2	O sistemu	5
2.1	Dejstva in izzivi	5
2.2	Sistem za upravljanje s spletnimi vsebinami	6
3	Razvojna orodja in tehnologije	13
3.1	Potrebna programska orodja	13
3.2	Tehnologije	15
4	Načrtovanje in razvoj sistema	21
4.1	Ideja in načrtovanje sistema	21
4.2	Podatkovna baza	22
4.3	Razvoj sistema	23
4.4	Namestitev sistema	60
5	Uporaba sistema in administracija	61
5.1	Diagram primerov uporabe	61
5.2	Skrbniški del sistema	64
5.3	Uredniški del sistema za upravljanje z vsebinami	71

KAZALO

6 Sklepne ugotovitve	75
6.1 Ovire	75
6.2 Predlogi izboljšav	76
6.3 Prednosti	77
 Seznam slik	 77
Seznam tabel	82
Seznam izsekov kod	83
A Predloga (Template) strani spletnega mesta CEF	95
B Vsebina (Content) strani Staff spletnega mesta CEF	99
C Prikaz strani nekega dogodka spletnega mesta CEF za na- prave z različnimi ločljivostmi zaslona	103

Seznam uporabljenih kratic

kratica	angleško	slovensko
AJAX	asynchronous javascript and xml	asinhroni javascript in xml
CMS	content management system	sistem za upravljanje vsebin
CSS	cascading style sheets	kaskadni stilski slogi
HTML	hyper text markup language	označevalni jezik
HTTP	hypertext transfer protocol	prenosni protokol
IRC	internet relay chat	spletni klepet
IS	information system	informacijski sistem
LESS	leaner css	predprocesorski jezik css
PHP	hypertext preprocessor	programski jezik php
SQL	structured query language	strukturni poizvedbeni jezik
TV	template variable for MODx	dodatno polje za MODx
...

Povzetek

Primer študije je bil razvoj novega, zmogljivejšega in fleksibilnejšega spletnega sistema, ki je zamenjal obstoječi, zastareli sistem. Obstoječa rešitev je že združevala dve spletni domeni, vendar z rastjo institucije ni bila več kos vsem njenim novim in posebnim zahtevam. Zaradi izkušenj na področju razvoja spletnih aplikacij, želje po razvoju ter učinkovitejšem nadgrajevanju znotraj institucije je prišlo do odločitve za lasten razvoj v lastni infrastrukturi z ogradjem, ki bi programerju ponudil več svobode in omogočil realizacijo morebitnih kasnejših kompleksnejših zahtev institucije na področju spletne predstavitve. Za doseg tega cilja je bilo izbrano ogrodje MODx. Z omenjenim ogradjem so bile razvite vse posebne komponente zalednega vmesnika za udobno urejanje podatkov, uporabnikom je bila dodana možnost hitrega urejanja dinamičnih vsebin preko uporabniškega vmesnika, administratorjem pa močno zaledno okolje za urejanje sistema in možnost popravkov ter nadgradnje sistema kar preko brskalnika.

Ključne besede: sistem za upravljanje spletnih vsebin, CMS, MODx, PHP, MySQL, Ext JS, odziven design, urejanje preko uporabniškega vmesnika, zaledni vmesnik, sistem z več spletnimi domenami.

Abstract

The case study of this thesis was an example of development of new, more flexible and more efficient web operating system that replaced the existing one which had already been outdated and did no longer meet the changing needs of the organization. With the growth of the organization, the existing system which was available under two domains became insufficient over time. This led to a decision to develop organization's own web infrastructure in a system that would be more flexible and efficient and would allow for any subsequent potentially more complex solutions in organization's efforts to enhance the quality and availability of information to its member and partner institutions and to all other relevant stakeholders. To achieve this objective the web content management system MODx was selected which allows for all backend components to be configured in a way that the user can edit and add entries fast and directly. In addition, the system's user-friendly nature allows its users to frontend edit dynamic contents in users' interface that has been created for this purpose. Its added value lies also in enabling the system administrators to manage upgrades through the browser.

Keywords: Web Content Management System, CMS, MODx, PHP, MySQL, Ext JS, responsive design, frontend editing, backend, multi-domain site.

Poglavje 1

Uvod

Danes skoraj ni več podjetja, ki ne bi imelo svoje spletne strani. Spletna podoba je trend in je hkrati tudi obraz podjetja, ki si ga vsak posameznik danes zlahka ogleda že kar preko pametnega mobilnega telefona. Največkrat je to tudi prvi korak, ki ga opravimo, ko želimo spoznati določeno podjetje. S stališa podjetja je zato že skoraj obvezno imeti svojo spletno predstavitev, saj uporabniki to že kar pričakujejo. Če spletne predstavitve ni na voljo, lahko uporabnik dobi vtis, da podjetje ni v koraku s časom, in se zaradi tega morda odloči za konkurenta, ki ima po možnosti celo takšno predstavitev, da na uporabnika napravi močan vtis.

Na trgu je tako vedno več sistemov za upravljanje spletnih vsebin (CMS - Content Management System). Nekateri služijo enostavnim predstavitev, drugi služijo zahtevnejšim rešitvam, spletnim trgovinam, nekateri pa so zaradi posebnih zahtev naročnika izdelani v celoti po meri. Zelo pogosto sistem temelji na programskem jeziku PHP. Med takšne najbolj poznane odprtokodne sisteme prištevamo WordPress, Joomla, Drupal, SilverStripe ter mnoge druge. Vendar to, kar je najbolj poznano, še ne pomeni nujno tudi najboljše. Pri nas se je pojavila zahteva, da poiščemo sistem, ki bi bil za urednike izredno enostaven in učinkovit, hkrati pa naj bi omogočal zahtevnejše rešitve, ko je to potrebno. Zaradi specifične zahteve podjetja nas niso zanimali najpogostejše uporabljene sistemi, pač pa sistemi, ki bi programerju omogočali

svobodo in fleksibilnost, ko bi bilo to potrebno, četudi bi morda bil na začetku časovni vložek nekoliko večji zaradi usvojitve potrebnega znanja o sistemu in zaradi spoznavanja njegovih zmožnosti. Po analizi zahtev in predvidevanja smo prišli do zaključa, da bi izbrani CMS moral biti ustrezen oziroma bi moral omogočati sledeče:

- je odprtokodni sistem;
- omogoča napredne in zmogljive možnosti za upravljanje s pravicami na uporabniških skupinah;
- omogoča enostavno upravljanje z dodatki oziroma komponentami drugih avtorjev ali kar lastnih (preko zalednega sistema za dodatke);
- razvoj lastnih komponent zalednega vmesnika z modernim izgledom in fleksibilnostjo;
- uporablja objektno-relacijski most (ORB - Object-relational Bridge);
- večkontekstno okolje (Multi Context Environment);
- sistemske in poljubne nove nastavitve na nivoju konteksta;
- možnost kreiranja novih razredov (Class) ali njihovih razširitev;
- enostavno dodajanje novih polj;
- možnost uporabe obrazcev v kombinaciji z zbirkami določenih nastavitvev ter spremenljivkami;
- možnost pisanja in uporabe kratkih PHP programov;
- možnost predpomnenja po meri;
- leksikone (Lexicons / Language);
- možnost urejanja kode preko brskalnika v zalednem vmesniku za administratorje v poljubnem HTML urejevalniku za brskalnike;

- upravljanje z datotečnim sistemom za dokumente urednikov in za programske datoteke za administratorje);
- možnost uporabe in razvoja Pluginov za izbrane systemske dogodke;
- močno podporno okolje, ki se odraža v pomoči uporabnikov in razvijalcev;
- enostavno integracijo poljubnega odzivnega ogrodja.

To in še več omogoča ogrodje MODx, ki je konec leta 2013 dobilo nagrado za People's Choice Winner for Best Open Source CMS [1], leta 2012 pa nagrado za Critic's Choice for Best Open Source CMS [2]. To je sistem, ki smo ga izbrali in uporabili tudi mi. Z njim smo uspešno nadomestili prejšnji zastarel in omejen sistem, uspešno smo migrirali še aktualne vsebine ter si omogočili možnost svobode pri realizaciji novih potreb spletne predstavitve podjetja.

Poglavje 2

O sistemu

Glavni namen sistema je enostavno in hitro upravljanje dinamičnih vsebin večkontekstnega spletnega sistema, kjer en kontekst predstavlja eno spletno mesto oziroma domeno. Konkretno gre za spletni sistem z domenama `http://www.cef-see.org` (v nadaljevanju **stran CEF**) in `http://www.crf.si` (v nadaljevanju **stran CRF**). Poudarek je na urejanju dinamičnih vsebin uporabniškega vmesnika, novic in dogodkov, na odzivnem designu, performansi ter možnosti razvoja fleksibilnih komponent zalednega vmesnika za poljubne vsebine, kot so galerija, publikacije, kontakti, spletna prijavnica ... To vse je bilo realizirano z ogrodjem MODx, ki omogoča vse že v Uvodu našete kvalitete.

2.1 Dejstva in izzivi

Pri načrtovanju smo se zavedali naslednjih dejstev in izzivov, za katere menimo, da so zelo pomembni:

- **DESIGN in STRUKTURA STRANI** ima na koncu za obiskovalca strani največjo težo. Uporabnik bo stran z veseljem še kdaj obiskal, če mu bo že na prvi pogled všeč. Največkrat je vstopna stran spletnega mesta kar prva stran, zato se moramo zavedati, da je to za nove obiskovalce morda celo najpomembnejša stran. Uporabnik se mora brez težav hitro znajti

na strani, prav tako je dobro priporočilo, da se obiskovalec do iskane informacije dokoplje v največ treh klikih. Poleg tega ne smemo pozabiti na izbiro fiksnega ali odzivnega spletnega designa. Ta izbira namreč posledično pomeni več dela za razvijalce in testerje, saj je potrebno glede na ločljivost naprave preveriti vsako stran za vsako standardno ločljivost. Na koncu je seveda trud poplačan. Vse več je mobilnih naprav in vse več uporabnikov posega po pametnih telefonih in brska po spletu, zato so strani, ki so razvite v odzivnem designu so zato zagotovo v prednosti pred fiksnimi. Design in struktura sta neodvisna od sistema, zato je ta razvoj lahko ločen oziroma ponekod tudi paralelen.

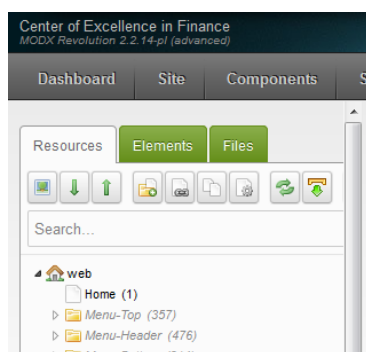
- **SISTEM:** Izbira sistemov za urejanje strani je zelo pestra, zato je včasih težko izbrati pravega. Po izbiri domnevno nekaj najbolj primernih sledi testiranje in zatem odločitev najprimernejšega. Sistem je izredno pomemben za programerje in urednike vsebine. Če uporabimo prispodobo, bi morda lahko sistem opisali kar takole: “Sistem je kot vozilo. Poleg voznika in sopotnikov je tudi od vozila odvisno, kako hitro in udobno bomo v različnih razmerah skupaj s sopotniki prispeli do želenega skupnega cilja.”

2.2 Sistem za upravljanje s spletnimi vsebinami

Osnovni vsebinski gradniki spletne strani, razvite s sistemom MODx Revolution, so **Resources**, **Elements** in **Files**.

2.2.1 Resource

Resource (stran) je objekt, do katerega lahko dostopamo preko URL naslova v brskalniku. Resource je lahko tipa **Document**, **Weblink**, **Symlink** in **Static Resource**. Privzeti tip je **Document**, ki predstavlja spletno stran sistema MODx [23].



Slika 2.1: Resources v zalednem vmesniku MODx.

2.2.2 Element

Elementi so objekti MODx-a, ki ustvarjajo, kontrolirajo, formatirajo, organizirajo in / ali vsebujejo vsebino. Poznamo glavne elemente (**Elements**) in ostale elemente ali značke (**Tags**). Predstavljeni so v tabelah. Vrednosti stolpca **element** in **značka** so zapisane v izvirnem jeziku (v angleščini).

element	opis	primer uporabe
Template	ni predstavitev (značke)	
Resource Field	[[*field]]	[[*pagetitle]]
Template Variable	[[*templatevar]]	[[*tags]]
Chunk	[\$chunk]	[\$header]
Snippet	[[snippet]]	[[getResources]]
Plugin	ni predstavitev (značke)	
Category	ni predstavitev (značke); so labele za organizacijo elementov	
Modul	ni predstavitev (značke); so komponente po meri	

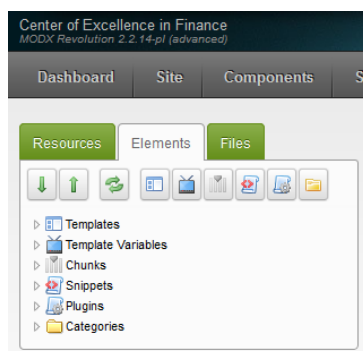
Tabela 2.1: Glavni elementi spletne strani MODx [20]

značka	opis	primer uporabe
Placeholder	[[+placeholder]]	[[+modx.user.id]]
Link	[[~link]]	[[~[[*id]]? &scheme='full']]
System Setting	[[++system_setting]]	[[++site_start]]
Language	[[%language_string_key]]	[[%LanguageStringKey? &language='en' &namespace='NamespaceName' &topic='TopicName']]
Comment		[[–this is a comment]]

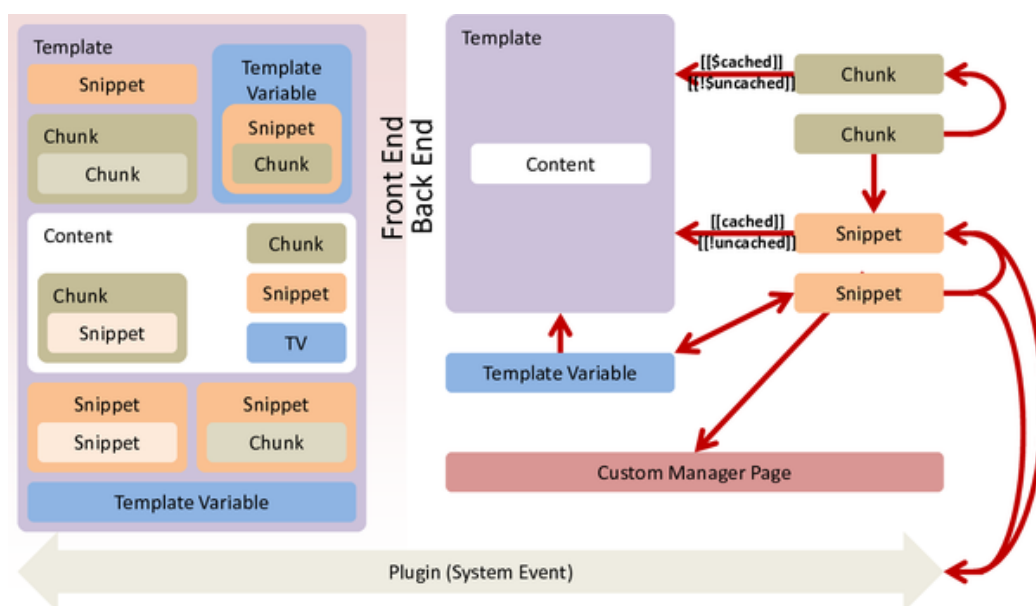
Tabela 2.2: Ostali elementi ali **značke** spletne strani MODx [20].

Glavne elemente lahko najdemo v drevesu elementov v zavihku **Elements** levo zgoraj zalednega vmesnika MODx. Elementi se med sabo lahko zelo prepletajo (gnezdenje elementov). Slika 2.3 prikazuje priporočena prepletanja glavnih elementov znotraj elementa Template, a se seveda vedno lahko najde še kakšna druga možnost.

Template (predloga) običajno vsebuje značke HTML in značke MODx, ki določajo postavitev in videz spletne strani. Ko se za ogled preko brskalnika zahteva določen dokument, MODx naloži dokument in njegovo predlogo



Slika 2.2: Glavni elementi v zalednem vmesniku MODx.



Slika 2.3: Priporočena prepletanja elementov znotraj Template [21].

ter najde vse značke MODx-a in jih nadomesti z ustreznimi vrednostmi iz dokumenta. Šele nato se brskalniku uporabnika pošlje končna stran [22].

Resource Field (polje strani) je lahko splošno ali nastavitveno in vsebuje določeno informacijo o strani. Splošna polja so: id, template, published, pagetitle, longtitle, description, introtext, alias, parent, link_attributes, menu_title, menuindex, hidemenu, content (slika 2.4). Nastavitvena pa so: isfolder, searchable, cacheable, createdby, editedby, deleted, deletedby, publishedby, createdon, publishedon, editedon, pub_date, unpub_date [23] (slika 2.5). Vrednosti skoraj vseh omenjenih polj določene strani lahko enostavno urejamo v zalednem vmesniku, če imamo za to ustrezne uporabniške pravice.

Template Variable / TV (dodatno polje strani) je polje, ki ga kreira uporabnik, ko potrebuje dodatno polje strani. TV torej predstavlja možnost razširitve seznama polj strani [24]. Ta polja so lahko poljubnega obstoječega tipa TV ali pa tipa, ki smo ga skreirali kar sami ali pa morda pridobili z instalacijo MODx Extra (dodatka) preko Package Managementa. Nekaj najpogostejših obstoječih tipov je: Date, Email, File, Hidden, Image, Listbox

Slika 2.4: Splošna polja strani.

(Multi-Select), Listbox (Single-Select), Number, Radio Options, RichText, Text, ResourceList, URL ...

Chunk je običajno statičen kos vsebine, ki jo lahko uporabimo na strani na več mestih. Ne vsebuje PHP kode, pač pa običajen tekst, ki je lahko tudi HTML ali morda JavaScript. Tak primer bi lahko bil glava ali noga strani. Poseben primer je, ko se **Chunk** uporablja kot **mini-template**, ki vsebuje poljuben tekst in **Placeholder** značke (**Placeholder**ji se obnašajo kot spremenljivke), lahko pa vsebuje tudi značke **Snippet**ov, ki so klici PHP izvedljive kode; te pa lahko ustvarjajo dinamične vsebine strani [25].

Snippet je kos izvedljive PHP kode. Večina jih je shranjenih v predpomnilniku, kar pomeni, da so shranjeni kot začasne, dinamične funkcije v predpomnilniku, a se jih lahko požene tudi v načinu brez predpomnenja. Tako kot Chunku, lahko tudi Snippetu določimo **Script Properties** (parametre). Ti parametri so lahko privzeti (na nivoju elementa ali v PropertySets), lahko pa se ob klicu določijo inline parametri, ki nadomestijo privzete. Poleg tega lahko znotraj Snippetov dostopamo do System Settings (sistemske nastavitve). Pravzaprav lahko znotraj Snippetov dostopamo do česarkoli in izvajamo karkoli. Snippet predstavlja glavni in najmočnejši razvojni element

Slika 2.5: Nastavitvena polja strani.

za razvijalce sistema [26]. Njihove klice lahko uporabimo kjerkoli (znotraj Templatea, Snippeteta, Chunka, Plugina) in se izvedejo, ko je stran obiskana.

Plugin je prav tako kos izvedljive PHP kode, vendar se z razliko od Snippeteta ta koda izvaja ob izbranih System Events (sistemskih dogodkih). To pomeni, da nam je omogočeno, da lahko na skoraj katerikoli točki procesiranja izvedemo željeno operacijo. Plugin je koda, ki posluša dogodke in se izvede, ko se sprožijo izbrani dogodki. Sistemskih dogodkov (uporabniških in zalednih) je preko 100. Plugine lahko enostavno onemogočimo ali omogočimo kar preko zalednega vmesnika, kar predstavlja neomejene možnosti in fleksibilnost pri doseganju poljubnih ciljev, ne da bi spreminjali jedro MODx-a. S tem je omogočeno tudi brezskrbno posodabljanje MODx jedra, saj je Plugin ločena koda od jedra MODx. Primeri nekaj dogodkov: OnWebPage-Prerender, OnWebLogin, OnDocFormSave, OnPageNotFound, OnUserSave ... [27].

Category (kategorija) je labela, ki jo lahko priredimo glavnemu elementu, kar nam pomaga pri organizaciji elementov, ko je teh veliko, ter nam omogoča tudi kontrolo dostopa. Mogoče je tudi gnezdenje kategorij.

Placeholder ni prikazan v drevesu elementov, vendar je kvalificiran kot ostali element, saj vsebuje vsebino, do katere se lahko dostopa preko **Placeholder** značke. V večini primerov se njihove vrednosti določajo znotraj Snippetov in Pluginov in ne preko vnosne maske zalednega vmesnika. **Placeholder**je lahko kličemo praktično kjerkoli [77].

Link je okrajšava za URL strani, ki bazira na polju Resourcea ID. URL določene strani lahko dobimo tako, da uporabimo tag **Link** `[[~##]]`, kjer je `##` ID strani. Namesto ID strani lahko uporabimo tag **MODx** `[[*id]]`, kar se zamenja z ID-jem trenutno dostopane strani (`[[~[[*id]]]]`).

System Setting. V zalednem vmesniku je v meniju **System Settings** mogoče poleg splošnih nastavitev strani nastaviti tudi nastavitve tega vmesnika, nastavitve datotečnega sistema, predpomnilnika, avtentikacije in varnosti ter še mnoge druge. Nastavitev je skupno okoli 200, lahko pa ustvarimo tudi nove, poljubne [28]. Do vseh nastavitev enostavno dostopamo kjerkoli v vseh elementih (**Elements**) in na vseh straneh (**Resources**) **MODx-a**. Vsako sistemsko nastavitve je mogoče tudi ponovno definirati na nivoju konteksta, kar daje programerju še dodatno fleksibilnost. Torej ima lahko zaledni vmesnik za ime strani (nastavitve `site_name`) vrednost npr.: “To je zaledni vmesnik,” uporabniški pa nekaj drugega, npr.: “To je uporabniški vmesnik.”

2.2.3 File

To so datoteke, ki poleg glavnih elementov prav tako predstavljajo del vsebine spletne strani. So največkrat slikovne datoteke, datoteke tipa CSS, JavaScript, TXT, PDF, DOCX, XLSX ter še mnoge druge. Te datoteke se nahajajo znotraj mape **assets**. Za vsako zaledno komponento gre ponavadi za mapo **assets/components/component_name** za datoteke, ki so dostopne preko spleta, ter mapo **core/components/component_name** za datoteke, ki ne smejo biti dostopne preko spleta (`component_name` pa je ime zaledne komponente). Splošna oziroma privzeta mapa za nalaganje dokumentov je mapa **assets/files**, za katero obstaja sistemsko nastavitve (`upload_files`), ki določa, kateri tipi datotek so dovoljeni za nalaganje.

Poglavje 3

Razvojna orodja in tehnologije

Za razvoj omenjenega sistema se uporabljajo različna orodja in tehnologije. V nadaljevanju bomo predstavili vsa orodja in tehnologije, ki smo jih uporabili pri razvoju našega spletnega sistema.

3.1 Potrebna programska orodja

3.1.1 Linux Ubuntu

Linux Ubuntu je odprtokodna distribucija Linuxa, ki temelji na distribuciji Debian. Ubuntu je operacijski sistem, ki je osredotočen na uporabnost, kar med drugim vključuje tudi orodje `sudo` za administrativna opravila in `APT` (Advanced Packaging Tool). Je primeren tako namizni rabi kot strežnikom [3].

3.1.2 Nginx

Nginx je odprtokodni, visoko zmogljivi spletni strežnik in obratni posredniški strežnik za HTTP, HTTPS, SMTP, POP3 ter IMAP protokole. Znan je po visoki zmogljivosti, fleksibilni konfiguraciji in majhni spominski porabi. Spletni strežnik skrbi za prejem zahtev, uporabniku spletnega mesta pa odgovarja z ustreznimi datotekami HTML [4]. Pogosto se uporablja za zelo obiskana

spletna mesta, kot so GitHub, Airbnb, WordPress.com ... [5]

3.1.3 PHP (prevajalnik)

PHP primarno teče na spletnem strežniku, kjer jemlje PHP izvorno kodo za vhod in generira spletno stran kot izhod [6].

3.1.4 PHP-FPM

PHP-FPM (FastCGI Process Manager) je enostaven in robusten odprtokodni procesni upravljalnik za PHP [7].

3.1.5 MySQL

MySQL je sistem za upravljanje s podatkovnimi bazami. MySQL je odprtokodna implementacija relacijske podatkovne baze, ki za delo s podatki uporablja jezik SQL. Deluje na principu odjemalec–strežnik. Za dostop do podatkovne baze MySQL obstaja veliko število odjemalcev, zbirk ukazov in programskih vmesnikov [8].

3.1.6 MODx Revolution

MODx Revolution je sistem za urejanje spletnih vsebin za tiste, ki bi radi resnično brezkompromisno zasnovo in izjemno uporabniško izkušnjo. To daje programerju in urednikom popoln nadzor nad spletno stranjo in vsebino s prožnostjo in razširljivostjo za prilagajanje spreminjajočim se potrebam [9].

3.1.7 xPDO

xPDO ali `OpenExpedio` je “lahka” odprtokodna knjižnica ORB (Object-relational Bridge), ki izkorišča novo sprejet standard za ohranitev integritete baze podatkov v PHP 5.1 + in PDO [10]. PDO je akronim za PHP Data Object (PHP podatkovni objekt) [11]. xPDO je del MODx jedra.

3.1.8 jQuery

jQuery je hitra, majhna in s funkcijami bogata knjižnica za JavaScript. Omogoča, da se z manj kode napravi več. jQuery je enostaven za uporabo in deluje v različnih brskalnikih [12].

3.1.9 Ext JS

Sencha Ext JS je JavaScript ogrodje za zahtevnejše **desktop** aplikacije z modernim izgledom, neprimerljivo združljivostjo za različne brskalnike, napredno MVC arhitekturo ter sodobnimi UI gradniki [13]. Ext JS je prav tako že vključen v sistem MODx.

3.1.10 Foundation

Foundation je napredno odzivno ogrodje za uporabniški vmesnik, ki je hiter, enostaven za uporabo in nezahteven za učenje [14]. Odzivna spletna stran se prilagaja glede na velikost okna brskalnika.

3.2 Tehnologije

3.2.1 HTML

HTML je označevalni jezik za izdelavo spletnih strani in predstavlja osnovo spletnega dokumenta. S HTML poleg prikaza dokumenta v brskalniku določimo tudi strukturo in semantični pomen delov dokumenta [15]. HTML je zaporedje ukazov, ki se imenujejo značke (tags). Te se vedno nahajajo med znakoma `<` in `>` kot npr. `<div>`.

```
1 <html>
2   <head><title>Naslov v okviru okna</title></head>
3   <body>Vsebina predstavitvene strani</body>
4 </html>
```

Izsek kode 3.1: Primer HTML strukture enostavne spletne strani.

3.2.2 CSS

CSS so podloge, predstavljene v obliki preprostega slogovnega jezika, ki skrbi za prezentacijo spletnih strani. Z njimi definiramo stil HTML elementov v smislu pravil, kako se naj ti prikažejo na strani. Določamo lahko barve, velikosti, odmike, poravnave, obrobe, pozicije in vrsto drugih atributov. Bistvo uporabe CSS je tudi ločitev strukture strani, ki jo podaja HTML jezik skupaj z vsebino, od njene predstavitve. S tem omogočimo lažje urejanje in dodajanje stilov ter poskrbimo za večjo preglednost HTML dokumentov. Prav tako zmanjšamo ponavljanja kode, saj omogočimo množici strani uporabo istih podlog [16].

```
1 body {
2     background-color: black;
3 }
4 h1 {
5     margin-left: 24px;
6     color: red;
7 }
```

Izsek kode 3.2: Primer enostavnega CSS.

3.2.3 PHP

PHP je programski jezik za strežniško uporabo oziroma za razvoj dinamičnih spletnih vsebin na osnovi podatkovne baze [6]. Dinamične vsebine lahko urejamo v sistemu za urejanje vsebin, kot so npr. novice, dogodki Ob spremembah vsebin preko sistema v bazi so te spremembe opazne v generirani HTML kodi, ki jo generiramo s PHP kodo. PHP je enostaven in zelo priljubljen programski jezik. PHP kodo pišemo med `<?php ?>` (začetni in končni tag).

```
1 <html>
2   <head>
3     <title>Naslov v okviru okna</title>
4   </head>
5   <body>
6     <?php
7       //prirejanje vrednosti spremenljivke
8       //to vrednost bi lahko zamenjali npr.
9       //z imenom uporabnika aplikacije
10      $name = 'Peter';
11      //izpis vrednosti spremenljivke $name
12      echo "Pozdravljeni v PHP, $name!";
13    ?>
14  </body>
15 </html>
```

Izsek kode 3.3: Primer enostavne PHP skripte, vključene v HTML.

3.2.4 JavaScript

JavaScript je objektni skriptni programski jezik, ki ga je razvil Netscape, da bi spletnim programerjem pomagal pri ustvarjanju interaktivnih spletnih strani. JavaScript lahko sodeluje s HTML kodo in s tem poživi stran z dinamičnim izvajanjem. Podpirajo ga vsi novejši spletni brskalniki. Koda se vključi v oznako `<script>`, s katero brskalniku povemo, da želimo vdelano kodo izvajati in ne prikazati [17].

```
1 <html>
2   <head>
3     <title>Naslov v okviru okna</title>
4   </head>
5   <body>
6     Lep je dan in...
7   </body>
8   <script>
9     // 'pop-up' okno z napisom
10    alert("tako bo tudi jutri!");
11  </script>
12 </html>
```

Izsek kode 3.4: Primer enostavne JavaScript skripte, vključene v HTML.

3.2.5 SQL

SQL ali strukturirani povpraševalni jezik za delo s podatkovnimi bazami je najbolj razširjen in standardiziran povpraševalni jezik za delo s podatkovnimi zbirkami in s programskimi stavki, ki posnemajo ukaze v naravnem jeziku. Najpogostejša operacija v SQL-u je poizvedba, ki se opravi s SELECT stavkom. SELECT stavek vrne podatke iz ene tabele ali več. Standardni SELECT stavek nima nobenega vpliva na podatke v podatkovni bazi in jih ne spreminja [18].

```
1 SELECT e.title
2 FROM event e
3 WHERE e.title LIKE '%focus%'
4 ORDER BY e.publishedon DESC
5 LIMIT 3;
```

Izsek kode 3.5: Primer enostavne strukture SELECT stavka.

3.2.6 AJAX

Ajax (asinhroni JavaScript in XML) je skupina medsebojno povezanih spletnih razvojnih tehnik, uporabljenih za ustvarjanje interaktivnih spletnih aplikacij. Z Ajaxom si lahko spletne aplikacije izmenjujejo podatke s strežnikom asinhrono v ozadju, ne da bi bilo potrebno ponovno nalagati strani, s čimer je omogočeno tekoče in hitrejše spremljanje ter spreminjanje vsebine na spletni strani. Podatki se prenašajo z objekti XMLHttpRequest [19].

```
1 <script>
2 var a_np_button_ids = [];
3 var spans = $('span.np_button');
4 spans.each(function() {
5     var id = parseInt($(this).attr('rel'), 10);
6     a_np_button_ids.push(id);
7     $(this).removeClass('hidden');
8 });
9 if (a_np_button_ids.length > 0) {
10     $.ajax({
11         "url": "getediticons.php",
12         "dataType": "json",
13         "type": "POST",
14         "data": {
15             "ids": a_np_button_ids
16         },
17         "success": function(json) {
18             $.each(json, function(id, htmlForm) {
19                 spans.filter('span[rel="' + id + '"]')
20                     .replaceWith(htmlForm);
21             });
22         },
23         "error": function(xhr, textStatus, error) {
24             console.log('ERROR: ' + error);
25         }
26     });
27 }
28 </script>
```

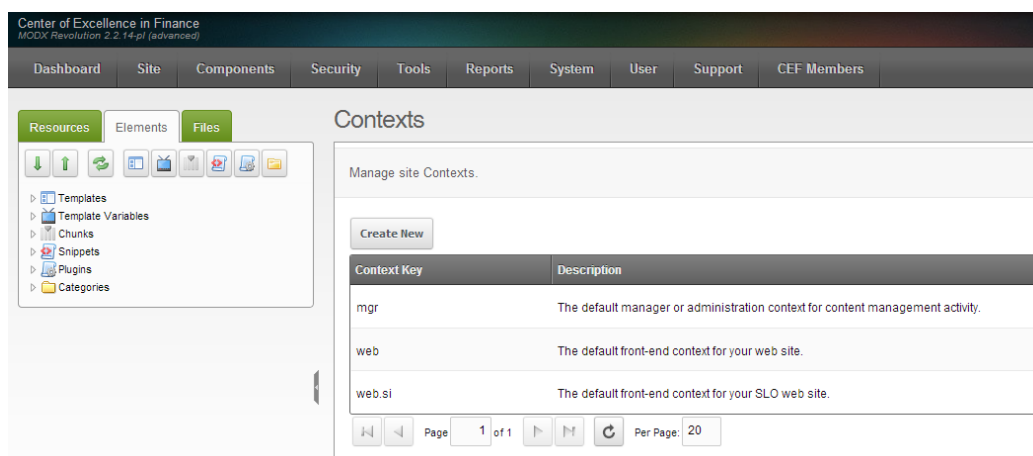
Izsek kode 3.6: Naš primer AJAX kode z uporabo knjižnice jQuery.

Poglavje 4

Načrtovanje in razvoj sistema

4.1 Ideja in načrtovanje sistema

Ideja je bila, razviti sistem, preko katerega bi bilo mogoče urejati vsebine trenutno vsaj za dve spletni mesti oziroma domeni. Vsako mesto bi imelo svoje nastavitve, template in vsebine. To se v MODx-u enostavno doseže tako, da se za vsako spletno mesto ustvari svoj kontekst (Context) v meniju zalednega vmesnika, kar prikazuje Slika 4.1.

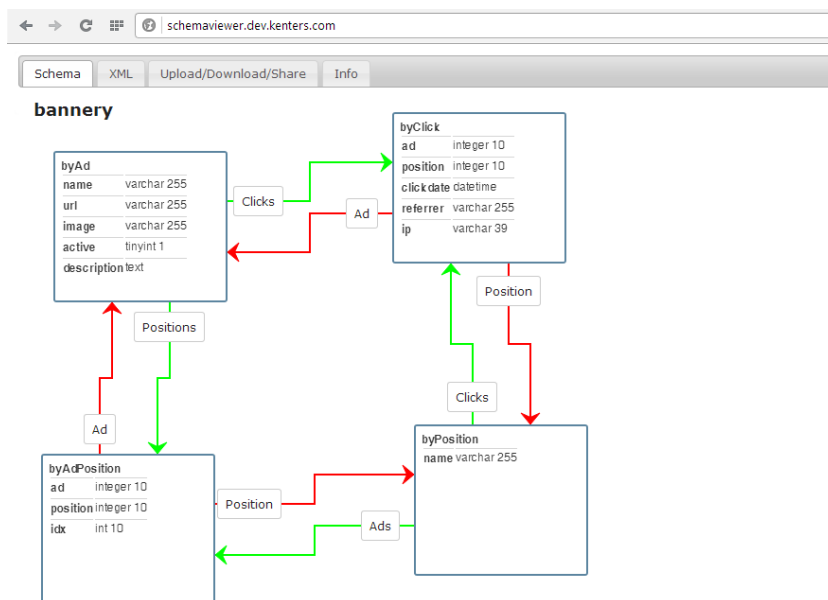


Slika 4.1: Seznam kontekstov v zalednem vmesniku.

Ker lahko znotraj zalednega vmesnika urejamo vsebino različnih kontekstov, pomeni, da lahko imamo skupne zaledne komponente. To je še ena funkcionalnost, ki jo ponuja MODx. Za pravilno delovanje uporabniškega vmesnika pa za preklap med konteksti ustrezno poskrbi poseben Plugin ob določenem sistemskem dogodku.

4.2 Podatkovna baza

Za delovanje MODx-a smo poleg drugih orodij morali namestiti tudi strežnik MySQL. MODx ob svoji namestitvi ustvari skupaj s tabelami svojo zbirko podatkov. Model podatkovne zbirke MODx predstavlja **shema** XML, ki se nahaja na lokaciji `core/model/schema/modx.mysql.schema.xml` znotraj MODx mape [29]. Za vizualizacijo **sheme** se lahko uporabi spletno orodje **xPDO Schema Visualizer** [30], ki pa se bolje obnese na krajših shemah XML (npr. za zaledne komponente, primer je Slika 4.2). Rdeča barva pomeni agregatno



Slika 4.2: xPDO Schema Visualizer: model za shemo zaledne komponente BannerY, ki jo uporabljamo v našem sistemu.

relacijo (aggregate relation), zelena pa kompozitno (composite relation) [70].

Podatki znotraj podatkovne zbirke MODx se običajno udobno urejajo preko objektov xPDO, katerih model PHP predstavlja shema XML.

4.3 Razvoj sistema

Kot večina razvojnih procesov sledi tudi razvoj IS določenemu življenjskemu ciklu, oziroma razvojnemu modelu, ki določa zaporedje faz razvoja. Razvojni modeli IS zajemajo v grobem analizo, načrtovanje in izvedbo, med seboj pa se razlikujejo predvsem po podrobnejši delitvi faz na aktivnosti in v zaporedju ter načinu njihovega izvajanja. Razvojni modeli, ki so značilni za strukturni razvoj IS, so: zaporedni model, iterativni model in inkrementalni model [71]. Mi smo izbrali inkrementalni model. Inkrementalni ali postopni model je zelo popularen model razvoja IS, pri katerem razdelimo celoten problem na podprobleme ali module, ki jih rešujemo posebej in neodvisno od ostalih. Aplikacijo tako razvijamo po delih, ki zajemajo le določen del funkcionalnosti zelenega sistema, vendar so obenem dovolj samostojni, da jih lahko vključimo v produkcijo. Z moduli sčasoma pokrijemo celotno funkcionalnost in zaključimo razvoj. Razvoj posameznih modulov jemljemo kot samostojne podprojekte [72]. Zaradi sprotnih sprememb smo se včasih vračali nazaj, sicer pa je razvoj potekal tekoče. Omenili bomo le pomembnejše dele razvoja.

4.3.1 Struktura strani

Struktura uporabniškega vmesnika spletnega mesta CEF (in CRF) je dokaj enostavna. Slika C.3 prikazuje strukturo, ki velja za vse strani. Predstavlja jo osnovni Template strani, ki je eden glavnih gradnikov in skupen vsem stranem (Priloga A). Tisto, kar je drugače pri vsaki strani, je VSEBINA (polje Content za objekt Resource; Slika C.3). Tudi vsebina strani lahko vsebuje mini-template, ki jih imenujemo Chunk.



Slika 4.3: Template in VSEBINA strani.

4.3.2 Prikaz spletne strani

Grafično podobo osnovnih tipov strani je za nas v obliki PDF dokumenta oblikovalo podjetje, ki za nas oblikuje tudi publikacije in ostalo tiskovino. Za razvijalce bi bilo bolj udobno, če bi grafično podobo prejeli še v obliki CSS in HTML datoteke, saj bi s tem nekoliko prihranili na času. Tako smo morali sami sprogramirati tudi CSS in definirati HTML strukturo (Template). Zaradi lažjega razvoja in vzdrževanja CSS kode smo uporabili LESS, za doseglo odzivne grafične podobe strani pa CSS ogrodje Foundation. Primere prikaza spletne strani za naprave z različnimi velikostmi zaslona za eno izmed strani dogodkov prikazuje Priloga C.

4.3.3 Dogodki

Ena izmed glavnih dejavnosti znotraj institucije so zagotovo dogodki, ki so največkrat delavnice kratkih izobraževalnih programov ali konference. Ker smo jih želeli izpostaviti, smo njihove najave prikazali že na vstopni strani spletne strani CEF, čeprav je seznam vseh prihajajočih dogodkov na voljo tudi na strani **events**. Potrebno je bilo razviti tudi spletni obrazec za prijavo na izbrani dogodek, za kar smo uporabili komponento FormIt in FormSave.

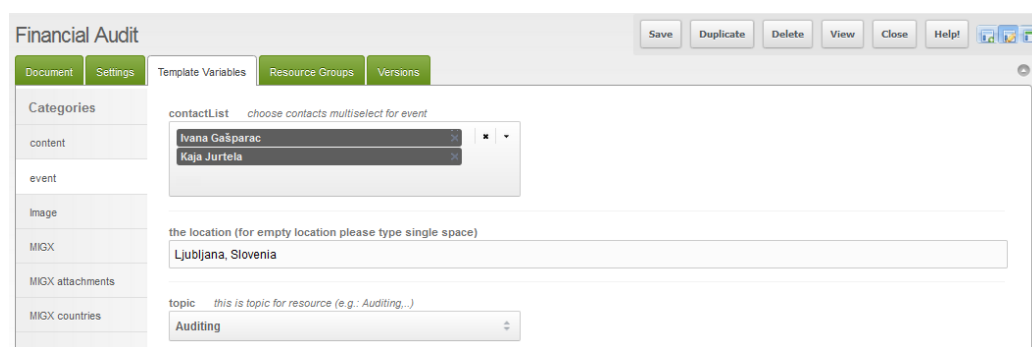
The screenshot shows the 'APPLICATION FORM' for the 'Center of Excellence in Finance'. The form is divided into three main sections:

- Event:** A dropdown menu showing 'Capital Budgeting (Sep 17–19, 2014)'. Below it, a note states: 'No fee is charged for officials working in the public sector.'
- Personal information:** Includes input fields for 'First name', 'Last name', and 'Middle name'. It also has fields for 'Birth country', 'Date of birth (dd/mm/yyyy)', and 'Nationality'. Gender is selected via radio buttons for 'Male' and 'Female'. A question 'Do you need VISA to enter Slovenia?' is followed by 'Yes' and 'No' radio buttons.
- Current job information:** Includes input fields for 'Job title' and 'Institution name'.

Slika 4.4: Spletna prijavnica za dogodke na strani CEF.

Zaradi želje, da bi bilo te dogodke mogoče urejati hitro in tudi za manj zahtevne uporabnike, smo urejanje preko uporabniškega vmesnika omogočili z Login in NewsPublisher komponentama ter nekaj JavaScript razširitvami. Vsebinsko, ki predstavlja odzivno stran dogodka, določajo naslednja polja: naslov, kraj dogodka, tema, posebna tema, datum začetka in datum zaključka dogodka, zadnji datum za prijavo, besedilo o dogodku, besedilo o predavateljih, seznam koordinatorjev dogodka ter seznam logotipov institucij predavateljev. Dodana so bila tudi polja, ki določajo, ali je dogodek samo za povabljenе, ali se vključi dogodek v seznam dogodkov v spletni prijavnici, ali pa onemogoči povezavo dogodka. Dogodek predstavlja objekt Resource tipa Document, ki je privzeti objekt za posamezno stran v MODx-u in ga je mogoče enostavno urejati preko zalednega vmesnika. Ker Reso-

urce nima posebnih polj, kot je npr. lokacija, smo za dogodke razširili seznam polj z objekti Template Variable (TV), ki so namenjeni prav temu. Tako smo v zalednem vmesiku ustvarili naslednje objekte TV: location, topicTV, specialTopicTV, start_date, end_date, contentChunkDisplay, applicationClosingDate, applyToIconTV, byInvitationOnlyIconTV, disableEvent, FE_about, FE_faculty, logoList, contactList. TV je mogoče uporabiti tudi za druge objekte. Polje tem se uporablja za prikaz sorodnih dogodkov, polje za posebno temo pa se uporablja za prikaz sorodnih dogodkov na eni izmed posebnih strani.

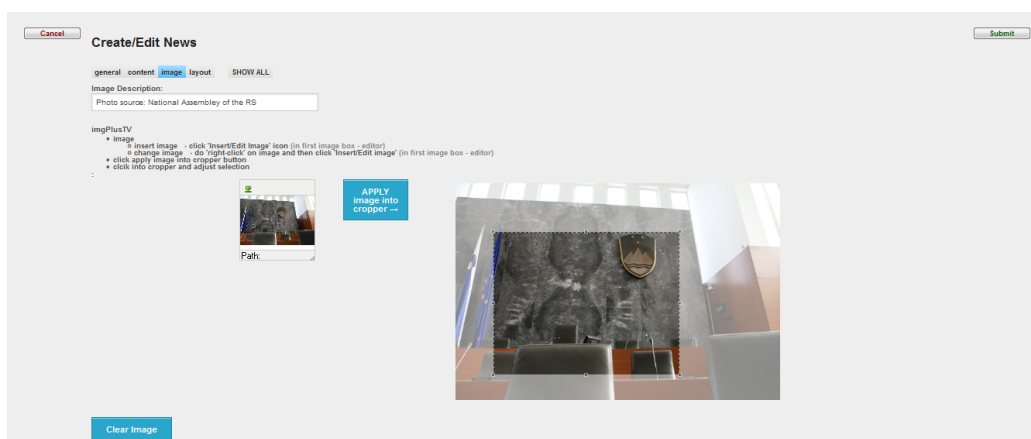


Slika 4.5: Urejanje Template Variables preko zalednega vmesnika za izbran dogodek.

4.3.4 Novice

Novice so ena izmed najbolj vitalnih delov spletne strani. Ker smo poleg dogodkov želeli izpostaviti tudi novice, smo jih umestili na vstopno stran ob dogodke. Prav tako smo ustvarili svojo stran news, ki predstavlja aktualne novice in arhiv starejših novic. Podobno kot pri dogodkih smo tudi novicam omogočili hitro urejanje preko uporabniškega vmesnika z Login in NewsPublisher komponentama ter nekaj JavaScript razširitvami. Vsebinsko, ki predstavlja odzivno stran novice, določajo naslednja polja: naslov, kraj dogodka, datum objave, posebna tema, vsebina novice in slika. Tudi za novice smo uporabili objekt Resource, tipa Document, za razširitev polj pa

obstoječe ali novo ustvarjene objekte Template Variable (TV): contentC-hunkDisplay, imageDescription, imgPlusTV. Posebnost je TV imgPlusTV, ki poleg informacije o izbrani sliki hrani še podatke, ki so potrebni za prikaz obrezane in pomanjšane slike na uporabniškem vmesniku. Posebnost je tudi polje introtext, ki ima vlogo vsebine novice za Newsletter. Za funkcionalnost obrezovanja slik preko uporabniškega vmesnika smo morali razširiti funkcionalnost komponente NewsPublisher. Uporabili smo knjižnico Jcrop [32] ter dodali nekaj lastne JavaScript kode.



Slika 4.6: Obrezovanja slike preko uporabniškega vmesnika za izbrano novico.

4.3.5 Galerija

Tudi galerija je zelo pomemben del spletne strani. Vanjo vedno in takoj po vsakem dogodku objavimo nekaj utrinkov, ki si jih obiskovalci dogodkov zelo radi ogledajo. Nekateri celo prosijo za slike v spomin. Ker delavnic na letni ravni ni malo in ker je zelo pomembno, da je galerija enostavna za uporabo, smo preverili skoraj vse možnosti za naš sistem. Na koncu smo izbrali MoreGallery komponento za MODx [43], ki ni brezplačna, a je vredna nakupa, saj je zelo učinkovita in enostavna za uporabo v zalednem vmesniku. Prikaz odzivne galerije na uporabniškem vmesniku smo morali razviti sami, za kar smo uporabili knjižnico Fotorama [33].



Slika 4.7: Uporabniški vmesnik galerije.

4.3.6 Statične strani

Razvitih je bilo kar nekaj statičnih strani in vsaka ima nekaj svojih posebnosti, zato smo jih pri razvoju obravnavali ločeno. Vsaka statična stran je predstavljala en manjši projekt. Po uskladitvi zahtev smo po načrtovanju te eno za drugo tudi realizirali.

4.3.7 Javne komponente








Sistem MODx preko menija Package Management omogoča iskanje, prenos, instalacijo, posodobitev in upravljanje javnih komponent drugih avtorjev. Prav tako je možno dodati nove Providerje, ki so repozitoriji za zaledne komponente MODx. Privzeti Provider je `modx.com`. Po namestitvi komponente so te ponavadi dostopne preko glavnega menija **Components**. Po pregledu naših zahtev in pred razvojem lastnih komponent po meri smo najprej preko menija Package Management raziskali, ali morda že obstaja kaj takega, kar bi potrebovali. Ugotovili smo, da lahko brez težav za doseg določenih ciljev uporabimo naslednje javne zaledne komponente:

- **BannerY:** To je preprosta komponenta za upravljanje zapisov z besedilom, sliko in URL povezavo v zalednem vmesniku. Na voljo je tudi Snippet za prikaz slik s povezavami na določenih položajih strani [36]. To komponento smo uporabili za upravljanje seznama institucij, s katerimi sodelujemo. Instituciji, ki predstavlja en zapis, lahko enostavno izberemo slikovni logotip, dopišemo URL naslov domače strani in izbiramo pozicije, ki jih je tudi mogoče dodajati in urejati. Ker smo pogrešali dodatno polje za opis institucije, smo komponento razširili in dodali novo tekstovno polje za opis. BannerY smo uporabili pri sekciji **Our partners** na prvi strani spletnega mesta CEF ter še na strani **our-partners**, kjer sicer opisujemo naše partnerje. Multi-select izbiro institucij partneric iz istega seznama pa smo uporabili pri urejanju posameznih strani CEF dogodkov (za CEF stran).

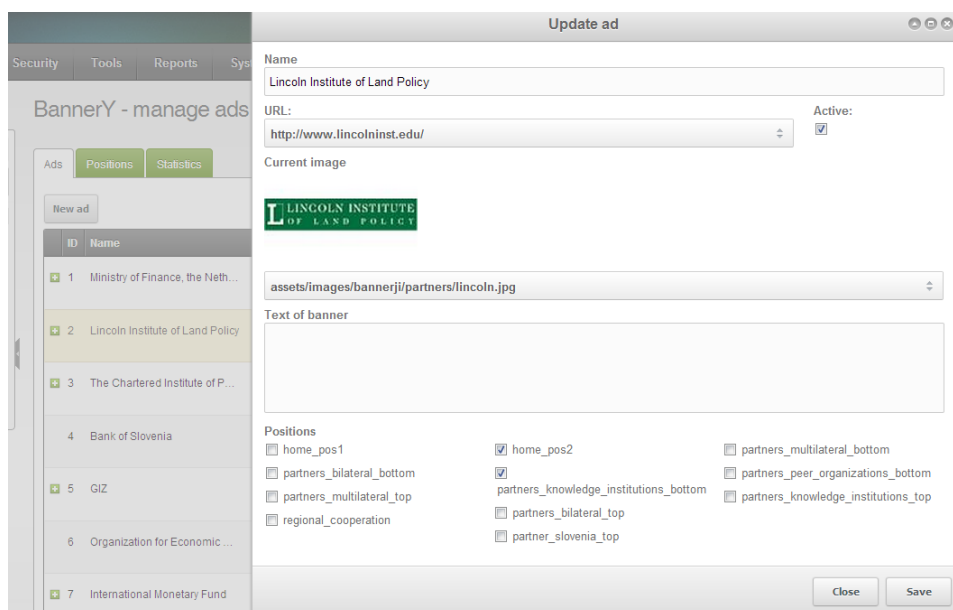
BannerY - manage ads

Ads Positions Statistics

New ad Select position Search Clear Filter

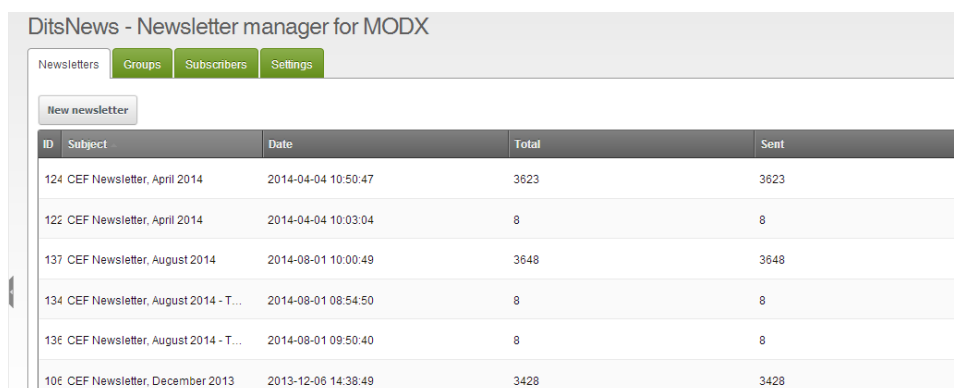
ID	Name	URL	Clicks	Active	Image
1	Ministry of Finance, the Neth...	http://www.government.nl/mi...	1568	Yes	
2	Lincoln Institute of Land Policy	http://www.lincolinst.edu/	1443	Yes	
3	The Chartered Institute of Pu...	http://www.cipfa.org.uk/	1465	Yes	
4	Bank of Slovenia	http://www.bsi.si/en/	1615	Yes	
5	GIZ	http://www.giz.de/en	1246	Yes	
6	Organization for Economic C...	http://www.oecd.org	1244	Yes	
7	International Monetary Fund	http://www.imf.org/external/in...	1308	Yes	

Slika 4.8: Seznam institucij v zaledni komponenti BannerY.



Slika 4.9: Urejanje posamezne institucije v zaledni komponenti BannerY preko seznama.

- DitsNews je učinkovita zaledna komponenta za upravljanje Newsletter e-sporočil za MODX Revolution. Omogoča upravljanje z naročniki in njihovo grupiranje, upravljanje posameznih grup, uvoz naročnikov z email naslovi, naročanje v seznam naročnikov preko obrazca na spletni strani ter možnost odjave naročnika preko URL povezave v Newsletterju [37]. DitsNews deluje tako, da zgenerira posamezna e-sporočila za naročnike izbranih skupin na podlagi že izdelanega Newsletterja kot strani (Resource). Za dinamično izgradnjo Newsletter strani smo morali napraviti svoj Template in napisati Snippet, ki na podlagi izbranih dogodkov in novic v sistemu za tekoči Newsletter zgenerira omenjeno Newsletter stran (Resource). Po kreaciji tega Resourcea in kreaciji Newsletterja v DitsNews za izbrane naročnike je sistem pripravljen za pošiljanje zgeneriranih Newsletter e-sporočil.



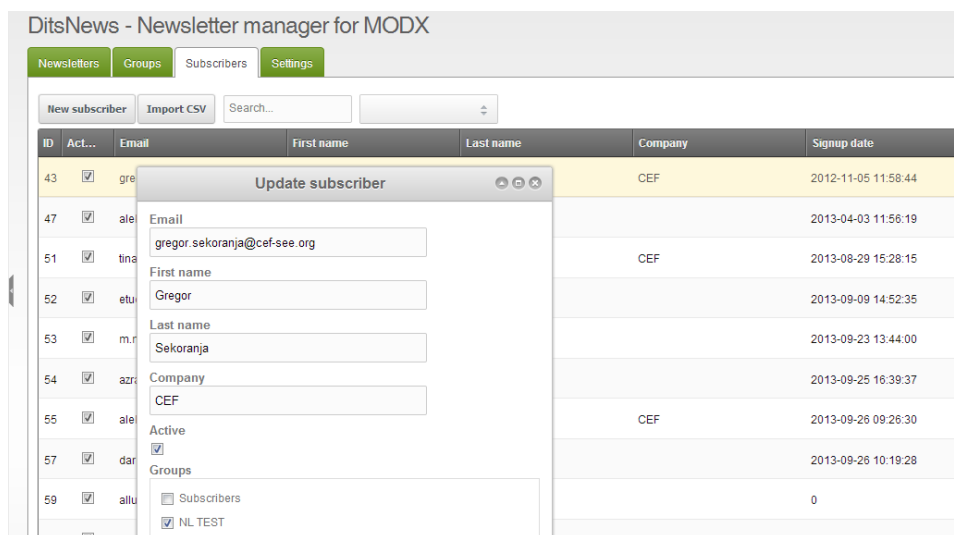
DitsNews - Newsletter manager for MODX

Newsletters Groups Subscribers Settings

New newsletter

ID	Subject	Date	Total	Sent
124	CEF Newsletter, April 2014	2014-04-04 10:50:47	3623	3623
122	CEF Newsletter, April 2014	2014-04-04 10:03:04	8	8
137	CEF Newsletter, August 2014	2014-08-01 10:00:49	3648	3648
134	CEF Newsletter, August 2014 - T...	2014-08-01 08:54:50	8	8
136	CEF Newsletter, August 2014 - T...	2014-08-01 09:50:40	8	8
106	CEF Newsletter, December 2013	2013-12-06 14:38:49	3428	3428

Slika 4.10: Seznam poslanih in neposlanih Newsletterjev v zaledni komponenti DitsNews.



DitsNews - Newsletter manager for MODX

Newsletters Groups Subscribers Settings

New subscriber Import CSV Search...

ID	Act...	Email	First name	Last name	Company	Signup date
43	<input checked="" type="checkbox"/>	gre...			CEF	2012-11-05 11:58:44
47	<input checked="" type="checkbox"/>	ale...				2013-04-03 11:56:19
51	<input checked="" type="checkbox"/>	tina...			CEF	2013-08-29 15:28:15
52	<input checked="" type="checkbox"/>	etu...	Gregor			2013-09-09 14:52:35
53	<input checked="" type="checkbox"/>	m.r...				2013-09-23 13:44:00
54	<input checked="" type="checkbox"/>	azr...				2013-09-25 16:39:37
55	<input checked="" type="checkbox"/>	ale...			CEF	2013-09-26 09:26:30
57	<input checked="" type="checkbox"/>	dar...				2013-09-26 10:19:28
59	<input checked="" type="checkbox"/>	allu...				0

Update subscriber

Email

First name

Last name

Company

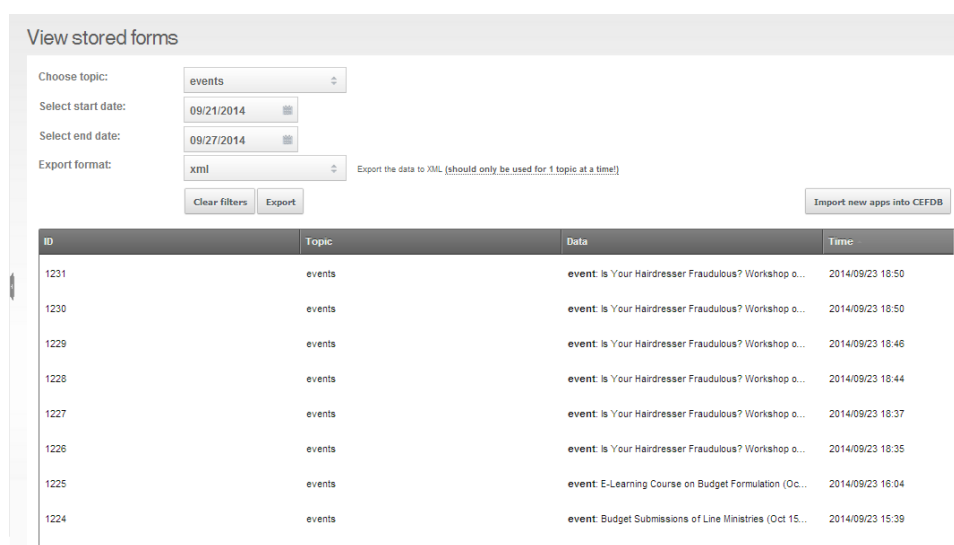
Active
☒

Groups
☐ Subscribers
☒ NL TEST

Slika 4.11: Urejanje naročnika v zaledni komponenti DitsNews preko seznama.

- FormSave je FormIt razširitev (FormIt Hook [38]), ki omogoča shranjevanje kateregakoli spletnega obrazca v podatkovno zbirko po oddaji obrazca. FormSave ima tudi zaledni vmesnik za pregledovanje prijav različnih spletnih obrazcev, omogoča pa tudi izvoz prijav v CSV ali XML. Možno je tudi kreirati lastne predloge za izvoz podatkov prijav [39]. FormIt smo uporabili za shranjevanje spletnih prijav na napovedane dogodke, ki so objavljeni na spletni strani CEF. Tudi v tem

primeru smo se lotili posebne razširitve, in sicer smo razvili skripto za uvoz podatkov prijavljenih oseb v drugo podatkovno zbirko, ki jo za delovanje uporablja druga, povsem ločena aplikacija, s čimer smo prihranili kar nekaj časa pri vnašanju novih podatkov v omenjeno interno aplikacijo. Akcijo uvoza predstavlja gumb Import new apps into CEFDB.



The screenshot shows a web interface titled "View stored forms". It includes a filter section on the left with the following options:

- Choose topic: **events** (dropdown)
- Select start date: **09/21/2014** (calendar icon)
- Select end date: **09/27/2014** (calendar icon)
- Export format: **xml** (dropdown)

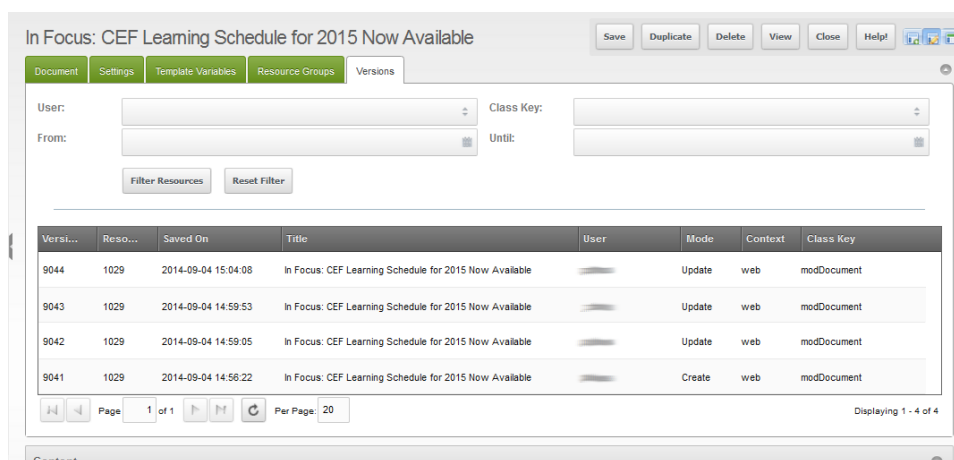
Below the filters are buttons for "Clear filters" and "Export". A note states: "Export the data to XML, (should only be used for 1 topic at a time!)". On the right, there is a button "Import new apps into CEFDB".

The main part of the interface is a table with the following data:

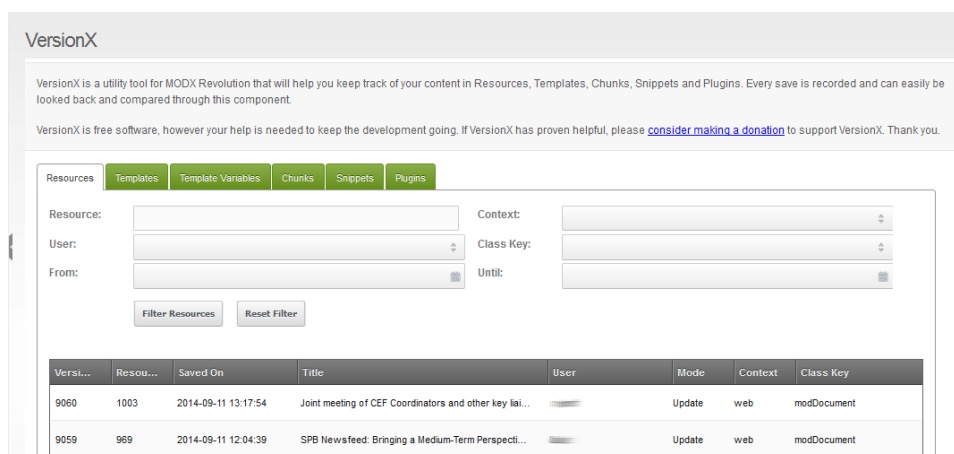
ID	Topic	Data	Time
1231	events	event: Is Your Hairdresser Fraudulous? Workshop o...	2014/09/23 18:50
1230	events	event: Is Your Hairdresser Fraudulous? Workshop o...	2014/09/23 18:50
1229	events	event: Is Your Hairdresser Fraudulous? Workshop o...	2014/09/23 18:46
1228	events	event: Is Your Hairdresser Fraudulous? Workshop o...	2014/09/23 18:44
1227	events	event: Is Your Hairdresser Fraudulous? Workshop o...	2014/09/23 18:37
1226	events	event: Is Your Hairdresser Fraudulous? Workshop o...	2014/09/23 18:35
1225	events	event: E-Learning Course on Budget Formulation (Oc...	2014/09/23 16:04
1224	events	event: Budget Submissions of Line Ministries (Oct 15...	2014/09/23 15:39

Slika 4.12: Seznam prijav različnih spletnih prijav v zaledni komponenti FormSave.

- VersionX je zaledna komponenta, ki za gradnike oziroma objekte Resource, Template, Chunk, Template Variable, Snippet in Plugin hrani vse prejšnje vrednosti in omogoča povrnitev objekta v prejšnje stanje. Ta funkcionalnost je zelo priporočljiva, saj predstavlja arhiv vsebinskih gradnikov. Frekvenca uporabe tega je v praksi zelo majhna, ampak zelo koristna.

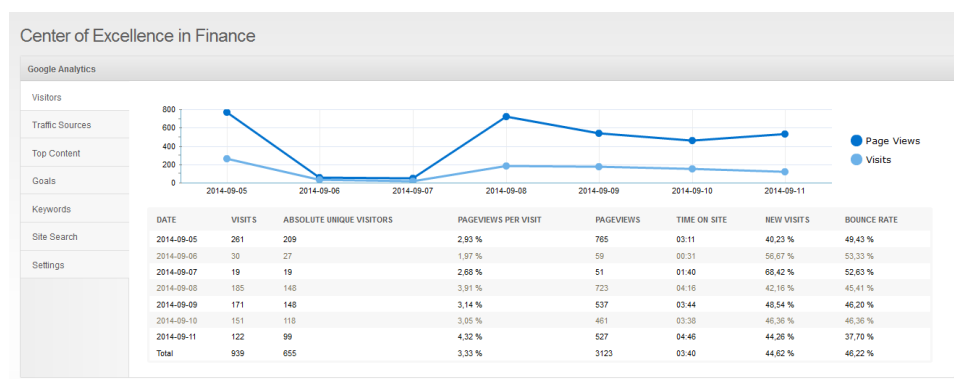


Slika 4.13: Če je Plugin VersionX aktiven, se pri urejanju preko zalednega vmesnika pojavi jeziček Versions, preko katerega se lahko opravi povrnitev.



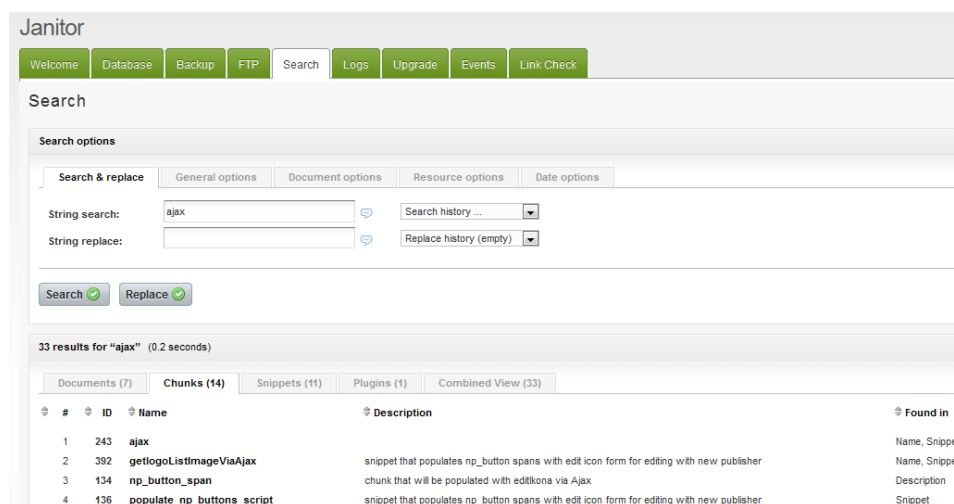
Slika 4.14: VersionX: celoten nadzor nad vsemi različicami glavnih vsebinskih objektov in možnost povrnitve, dostopno preko menija.

- Google Analytics Dashboard Widget je komponenta v nadzorni plošči, ki se ponavadi pojavi takoj po zaledni prijavi in prikazuje statistiko Google Analytics. Za uporabo tega Plugina je potrebno imeti račun Google Analytics [40], potrebno pa je tudi nastaviti kodo za spletno sledenje, kar se lahko doseže z uporabo dodatka Analytics [41].



Slika 4.15: Google Analytics Dashboard Widget: prikaz statistike Google Analytics po zaledni prijavi.

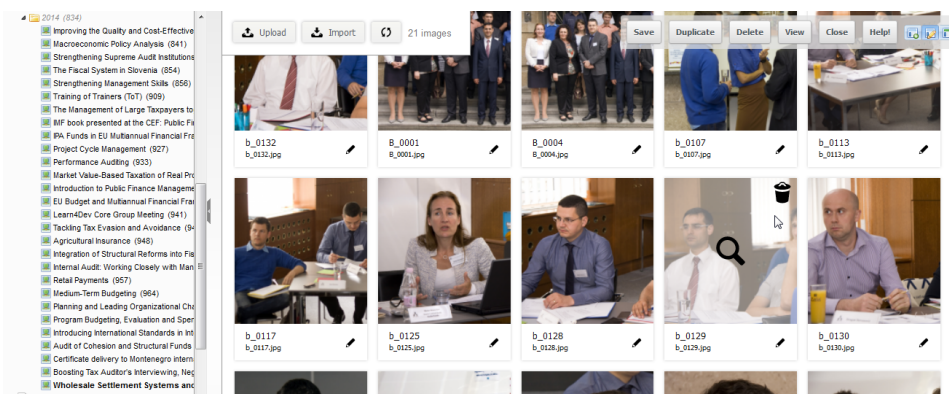
- Janitor je komponenta za MODx Revolution, ki je zelo uporabna za programske administratorje. Za nas je bila ta komponenta najbolj uporabna kot iskalnik glavnih vsebinskih gradnikov po ID, vsebini, imenu ali opisu [42].



Slika 4.16: Janitor: odličen pri iskanju po vsebinskih gradnikih znotraj MODx Revolution.

- MoreGallery je enostavna in učinkovita zaledna komponenta za kreiranje fotogalerije v MODx Revolution. Z udobnim drag'n'drop izbranih slik iz raziskovalca v zaledni vmesnik se enostavno naložijo slike v datotečni sistem MODx za izbrani Resource (tipa Gallery) [43].

Edino, kar manjka MoreGallery, je uporabniški vmesnik za odzivni design. Z nekaj truda smo razvili tudi tega.



Slika 4.17: MoreGallery: enostavno nalaganje slik, reorganizacija slik, možnost opisa za vsako sliko.

Obstajajo tudi nekatere preprostejše javne komponente, ki nimajo zalednega vmesnika, vsebujejo pa lahko zelo uporaben Snippet ali Plugin. Mi smo našli, instalirali, stestirali in na koncu tudi uporabili naslednje:

komponenta	opis
Analytics	Analytics je Snippet, ki vstavi kodo za sledenje Google Analytics storitve (Universal analytics.js) in / ali Google Analytics (ga.js) na vsaki strani. Iz statistike je mogoče izključiti promet od uporabnikov, ki so prijavljeni v kateremkoli kontekstu [41].
CodeMirror	CodeMirror je Plugin za MODx Revolution za sintaktično obarvanje kode za glavne elemente in strani MODx-a [44].

komponenta	opis
FormIt	FormIt je Snippet za dinamično procesiranje spletnih obrazcev (v HTML so to značke form) za MODx Revolution. Njegova naloga je validacija poslanih podatkov preko spletnega obrazca po njegovi oddaji; za tem lahko opravlja tudi nadaljnje akcije, kot je npr. pošiljanje e-pošte. FormIt ne generira spletnega obrazca, lahko pa ga napolni s poslanimi podatki in izpiše napake, če validacija podatkov ne uspe [45].
getCache	getCache je generični Snippet za predpomnenje vrnjenega izhoda kateregakoli glavnega elementa z uporabo nastavljivega handlerja za predpomnilnik [46].
getCollection	getCollection je Snippet, ki vrača polje xPDO objektov (xPDOObjects) z uporabo določenega xPDO kriterija (xPDOCriteria) ali pa vrača prazen niz, če je rezultat prazen [47].
getPage	getPage je generični Snippet za predstavitev ali navigacijo večstranskih pogledov za katerikoli glavni element [48].
getResourceField	getResourceField je enostaven Snippet, ki vrača vrednost izbranega polja strani (Resource Field) ali vrednost Template Variable za izbrano stran glede na podani ID [49].
getResources	getResources je generičen Snippet za prilagojen izpis seznama izbranih strani (Resources) glede na podane parametre [50].

komponenta	opis
iCalX	iCalX je paket Snippetov za ustvarjanje iCalendar datotek za prenos ter za ustvarjanje povezav “dodaj v koledar” za trenutno obiskano stran (Resource) [51].
Login	Login je paket Snippetov za MODx Revolution, ki omogoča prijavo preko uporabniškega vmesnika, posodobitev profila, registracijo in funkcionalnost “Pozabil sem geslo” [52].
MultiUploadDialog	MultiUploadDialog je Plugin, ki zamenja enostavno okno za nalaganje datotek v datotečni sistem MODx (Files) z oknom, ki omogoča izbiro več datotek naenkrat ter funkcionalnost “drag’n’drop” (povleci in spusti) [53].
NewsPublisher	NewsPublisher je paket Snippetov, ki omogoča in predstavlja prilagodljiv spletni obrazec za ustvarjanje novih strani (Resource) in urejanje obstoječih strani preko uporabniškega vmesnika. Za to komponento smo samoiniciativno tudi razvili in skupnosti vrnili uporabno razširitev NewsPublisher Tabs , ki omogoča organizacijo obstoječih polj vnosne maske v poljubne jezičke (Tabs) [54]. To smo realizirali z dodatkom JavaScript in CSS brez kakršnih koli sprememb Snippetov.
phpThumbOf	phpThumbOf je Snippet oziroma Output Filter [55], ki z uporabo knjižnice phpThumb na podlagi izbranih parametrov in URL slike ustvari novo sliko ter vrača URL slike. [56].
Rowboat	Rowboat je Snippet za iteracijo po vrsticah katerekoli tabele podatkovne zbirke MySQL in prilagojen izpis na podlagi izbranih parametrov. [57].

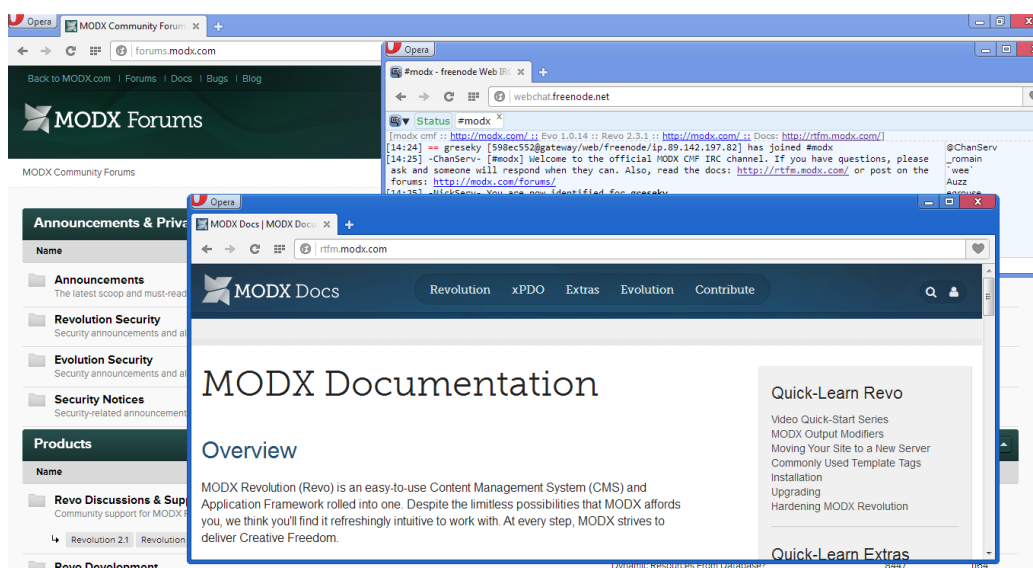
komponenta	opis
SimpleSearch	SimpleSearch je paket Snippetov, ki predstavlja enostaven iskalnik za MODx Revolution za uporabniški vmesnik [58].
translit	translit je razširitev MODx Revolution jedra, ki omogoča prilagodljiv transliteracijski servis. Ob namestitvi tega dodatka se pojavijo nove sistemske nastavitve (System Settings) pod kategorijo core → Friendly URL za transliteracijo znakov polja alias za objekt stran (Resource) [59].
Image+	Image+ je tip spremenljivke Template Variable , ki omogoča funkciji <i>obreži sliko</i> in <i>spremeni njeno velikost</i> brez spremembe originalne slike [60]. Obrezano sliko je potem mogoče uporabiti na uporabniškem vmesniku.
Wayfinder	Wayfinder je Snippet, ki lahko preišče določen del drevesne strukture strani (Resources), najde vse objekte, ki izpolnjujejo kriterije na podlagi izbranih parametrov, in izpiše seznam teh strani v poljubni obliki. [61].

Tabela 4.1: Seznam preprostih javnih zalednih komponent oziroma dodatka brez grafičnega vmesnika, ki smo jih namestili v naš sistem preko Package Managementa in jih tudi pridno uporabili.

Javne komponente MODx (Extras) je mogoče poiskati tudi na zelo uporabni spletni povezavi MODx Extras [31]. Ker vsega ni bilo mogoče rešiti z že obstoječimi javnimi rešitvami, smo določene komponente morali razširiti, določene pa smo morali razviti sami.

4.3.8 Lastne komponente

Zaradi specifične zahteve smo določene zaledne komponente morali razviti sami. V veliko pomoč nam je bila spletna dokumentacija MODx [62], kjer smo lahko dobili skoraj vse odgovore na začetna in tudi kasnejša vprašanja. V primeru večjih izzivov smo se lotili brskanja foruma MODx [63], kjer so odgovori na konkretne težave, s katerimi se srečujejo razvijalci v MODx. Kadar tudi to ni zadoščalo, smo izkoristili spletni klepet (IRC) kanala `modx` na mreži `freenode` IRC [64], kjer so na voljo programerji MODx s celega sveta. Kasneje smo skovali dobre stike z določenimi MODx uporabniki in celo programerji MODx sistema. Pomoč, ki smo jo dobili od te izjemne skupnosti, pridno vračamo in pomagamo tistim, ki so sedaj tam, kjer smo na začetku bili mi.

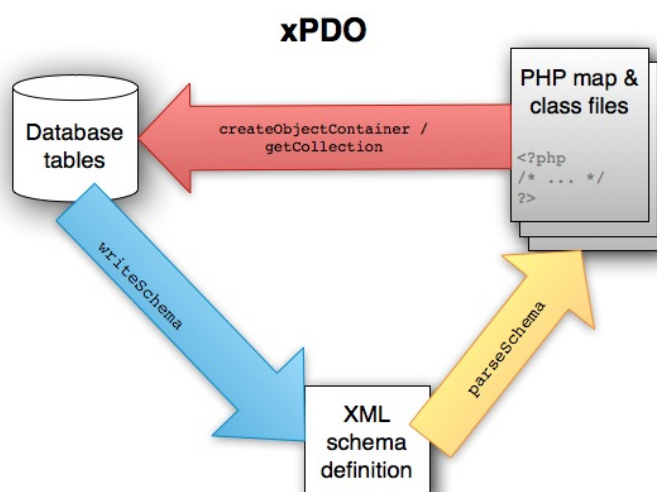


Slika 4.18: Razvoj lastnih komponent: pomoč na vsakem koraku.

Nepogrešljiv je bil tudi iskalnik Google [65], ki je kot vedno odličen asistent pri iskanju ostalih programerskih odgovorov. Proces razvoja komponent, ki smo jih razvili sami, bomo najpodrobneje prikazali na naši prvi komponenti, `Contacts`.

Contacts

je zaledna komponenta, ki je skupna za obe spletni mesti, CEF in CRF, predstavlja tudi osnovo za vse ostale komponente, ki smo jih razvili. Pri razvoju celotne komponente nam je bila v veliko pomoč tudi spletna dokumentacija MODx za razvoj zalednih komponent [66, 67]. Izpostavili bomo predvsem izdelavo `modela` za to komponento z uporabo xPDO sheme XML (v nadaljevanju samo shema) za podatkovno zbirko MySQL. Podrobna dokumentacija o izdelavi `modela` (kreiranju sheme ter generiranju `modela`) je prav tako na spletni dokumentaciji MODx [68]. Možno je tudi kreiranje `modela` iz obstoječih tabel v podatkovni zbirki s tehniko reverznega inženiringa, kar je prav tako podrobno opisano na spletni dokumentaciji MODx [69]. Smer razvoja je lahko poljubna, kar predstavlja Slika 4.19. Za komponento Contacts



Slika 4.19: Razvoj modela: smer razvoja je poljubna [69].

je shema precej preprosta, saj predstavlja le eno tabelo `contacts` in njena polja. Relacij ni bilo potrebno definirati [70].

```

1 <?xml version="1.0" encoding="UTF-8"?>
2 <model package="contacts" baseClass="xPDOObject" platform="mysql"
  defaultEngine="MyISAM" version="1.1">

```

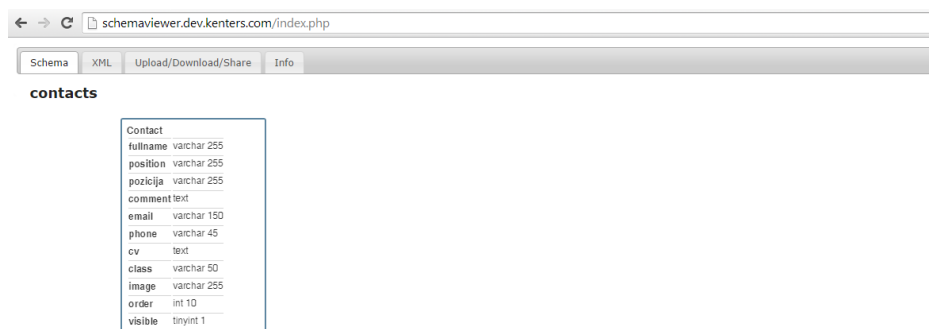
```

3 <object class="Contact" table="contacts" extends="xPDOSimpleObject">
4   <field key="fullname" dbtype="varchar" precision="255" phptype="string"
      null="false" />
5   <field key="position" dbtype="varchar" precision="255" phptype="string"
      null="false" />
6   <field key="pozicija" dbtype="varchar" precision="255" phptype="string"
      null="false" />
7   <field key="comment" dbtype="text" phptype="string" null="true" />
8   <field key="email" dbtype="varchar" precision="150" phptype="string"
      null="false" />
9   <field key="phone" dbtype="varchar" precision="45" phptype="string"
      null="true" />
10  <field key="cv" dbtype="text" phptype="string" null="true" />
11  <field key="class" dbtype="varchar" precision="50" phptype="string"
      null="true" />
12  <field key="image" dbtype="varchar" precision="255" phptype="string"
      null="true" />
13  <field key="order" dbtype="int" precision="10" attributes="unsigned"
      phptype="integer" null="true" />
14  <field key="visible" dbtype="tinyint" precision="1" phptype="integer"
      null="false" />
15 </object>
16 </model>

```

Izsek kode 4.1: Shema za Contacts komponento.

Vizualizacijo sheme prikazuje Slika 4.20, zgenerirana s spletnim orodjem xPDO Schema Visualizer [30].



Slika 4.20: Vizualizacija sheme z orodjem xPDO Schema Visualizer za komponento Contacts.

Na podlagi izdelane sheme se z `build skripto` [73] zgenerira model v PHP; to so xPDO razredi za objekte PHP in njihova preslikavanja (classes and maps) [66].

```

1 <?php
2 $xpdo_meta_map['Contact']= array (
3     'package' => 'contacts', 'table' => 'contacts',
4     'fields' =>
5         array (
6             'fullname' => NULL, 'position' => NULL, 'pozicija' => NULL, 'comment'
              => NULL, 'email' => NULL, 'phone' => NULL, 'cv' => NULL,
              'class' => NULL, 'image' => NULL, 'order' => NULL, 'visible' =>
              NULL
7         ),
8     'fieldMeta' =>
9         array (
10            'fullname' => array ('dbtype' => 'varchar', 'precision' => '255',
              'phptype' => 'string', 'null' => false),
11            'position' => array ('dbtype' => 'varchar', 'precision' => '255',
              'phptype' => 'string', 'null' => false),
12            'pozicija' => array ('dbtype' => 'varchar', 'precision' => '255',
              'phptype' => 'string', 'null' => false),
13            'comment' => array ('dbtype' => 'text', 'phptype' => 'string',
              'null' => true),
14            'email' => array ('dbtype' => 'varchar', 'precision' =>
              '150', 'phptype' => 'string', 'null' => false),
15            'phone' => array ('dbtype' => 'varchar', 'precision' => '45',
              'phptype' => 'string', 'null' => true),
16            'cv' => array ('dbtype' => 'text', 'phptype' =>
              'string', 'null' => true),
17            'class' => array ('dbtype' => 'varchar', 'precision' => '50',
              'phptype' => 'string', 'null' => true),
18            'image' => array ('dbtype' => 'varchar', 'precision' => '255',
              'phptype' => 'string', 'null' => true),
19            'order' => array ('dbtype' => 'int', 'precision' => '10',
              'attributes' => 'unsigned', 'phptype' => 'integer', 'null' =>
              true),
20            'visible' => array ('dbtype' => 'tinyint', 'precision' => '1',
              'phptype' => 'integer', 'null' => false)
21        )
22 );

```

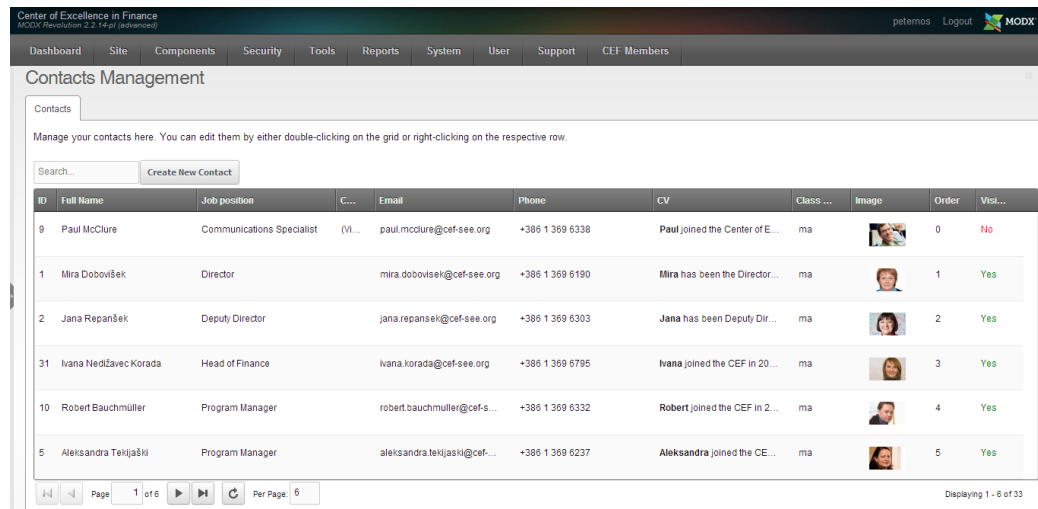
Izsek kode 4.2: To je zgeneriran `contact.map.inc.php`, ki je preslikava objekta Contract iz sheme modela v PHP model.

Dobljeni model PHP nam je omogočil programiranje z xPDO objektom Contact.

```
1 <?php
2 $classKey = 'Contact';
3 $packageName = 'contacts';
4
5 //nalaganje modela
6 $modx->addPackage($packageName, $modx->getOption('core_path') .
    'components/' . $packageName . '/model/');
7
8 //to je query
9 $criteria = $modx->newQuery($classKey);
10 $criteria->where(array('visible' => 1));
11 $criteria->sortBy('order');
12
13 //to je polje objektov 'Contact'
14 $contacts = $modx->getCollection($classKey, $criteria);
15
16 //vrnemo niz kontaktov, locenih z vejico
17 $aOutput = array();
18 foreach ($contacts as $contact){
19     $aOutput[] = $contact->get('fullname');
20 }
21 return implode(', ', $aOutput);
```

Izsek kode 4.3: Šolski primer Snippet, ki vrača kontakte, ločene z vejico, kot niz.

Objekte lahko, poleg izpisovanja, tudi ustvarjamo, urejamo in brišemo z uporabo Snippetov, kar je preprosto za programerje. Ker pa smo želeli, da bi s kontakti upravljali tudi običajni uporabniki, in sicer udobno preko uporabniku prijaznega zalednega vmesnika, smo morali razviti tudi tega. Razvili smo ga po navodilih spletne dokumentacije MODx [66] z ogrođjem Ext JS ter dodali še nekaj sprememb in lastne logike, kot je možnost reorganizacije kontaktov kar preko zalednega vmesnika. Za nekaj trikov v Ext JS smo izvedeli tudi s programsko kodo že nameščenih javnih komponent, dober vir učenja pa je lahko tudi koda jedra MODx. Zaledni vmesnik MODx ba-



Slika 4.21: Contacts: lastna zaledna komponenta: za udobno upravljanje s kontakti za obe spletni mesti.

zira na ogrodju Ext JS, ki ponuja skoraj neomejene možnosti na področju vmesnikov, vendar pa je programiranje v Ext JS lahko težavno, če ga še ne poznamo dovolj dobro. Sicer pa MODx dopušča programerju popolno svobodo pri razvoju zalednega vmesnika, kar pomeni, da lahko programer izbere tudi kakšno drugo ogrodje ali pa sploh nobenega.

```

1 Contacts.grid.Contacts = function(config) {
2     config = config || {};
3     Ext.applyIf(config,{
4         id: 'contacts-grid-contacts'
5         ,url: Contacts.config.connectorUrl
6         ,baseParams: { action: 'mgr/contact/getList' }
7         ,save_action: 'mgr/contact/updateFromGrid'
8         ,fields: ['id', 'fullname', 'position', 'pozicija', 'comment',
9                 'email', 'phone', 'cv', 'class', 'image', 'order', 'visible']
10        ,paging: true
11        ,pageSize: 50
12        ,autosave: true
13        ,remoteSort: true
14        ,anchor: '97%'
15        ,autoExpandColumn: 'order'
16        ,columns: [

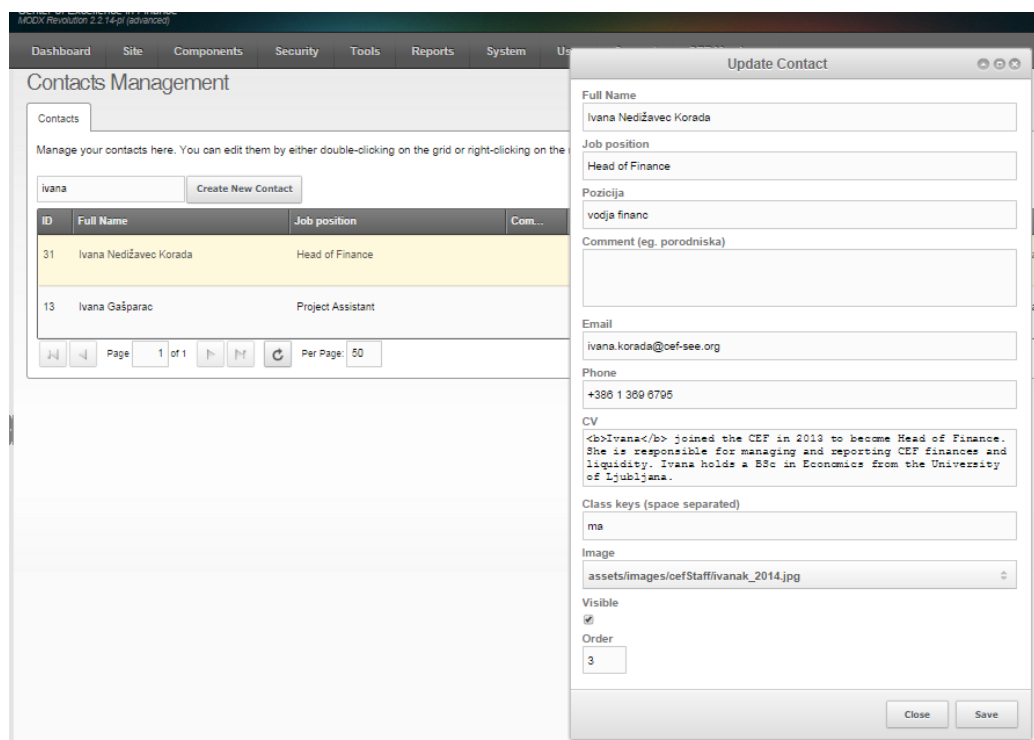
```



```
16     {header: _('id'), dataIndex: 'id', sortable: false, width: 60}
17     ,{ header: _('contacts.fullname'), dataIndex: 'fullname', sortable:
18         false, width: 350, editor: { xtype: 'textfield' }}
19     ,{ header: _('contacts.position'), dataIndex: 'position', sortable:
20         false, width: 350, editor: { xtype: 'textfield' }}
21     ,{ header: _('contacts.comment'), dataIndex: 'comment', sortable:
22         false, width: 100, editor: { xtype: 'textfield' }}
23     ,{ header: _('contacts.email'), dataIndex: 'email', sortable: false,
24         width: 350 , editor: { xtype: 'textfield' }}
25     ,{ header: _('contacts.phone'), dataIndex: 'phone', sortable: false,
26         width: 350, editor: { xtype: 'textfield' }}
27     ,{ header: _('contacts.cv'), dataIndex: 'cv', sortable: false, width:
28         350, editor:{ xtype: 'textfield' }}
29     ,{ header: _('contacts.class'), dataIndex: 'class', sortable: false,
30         width: 150, editor: { xtype: 'textfield' }}
31     ,{ header: _('contacts.image'), dataIndex: 'image', renderer:
32         {fn:function(img) {return renderGridImage(img,30)}}, sortable:
33         false, width: 200, editor: { xtype: 'textfield' }}
34     ,{ header: _('contacts.order'), dataIndex: 'order', sortable: true,
35         width: 120, editor: { xtype: 'textfield' }}
36     ,{ header: _('contacts.visible'), dataIndex: 'visible', sortable:
37         false, renderer: this.renderBoolean, width: 120}
38     ]
39 });
40 Contacts.grid.Contacts.superclass.constructor.call(this,config)
41 };
```

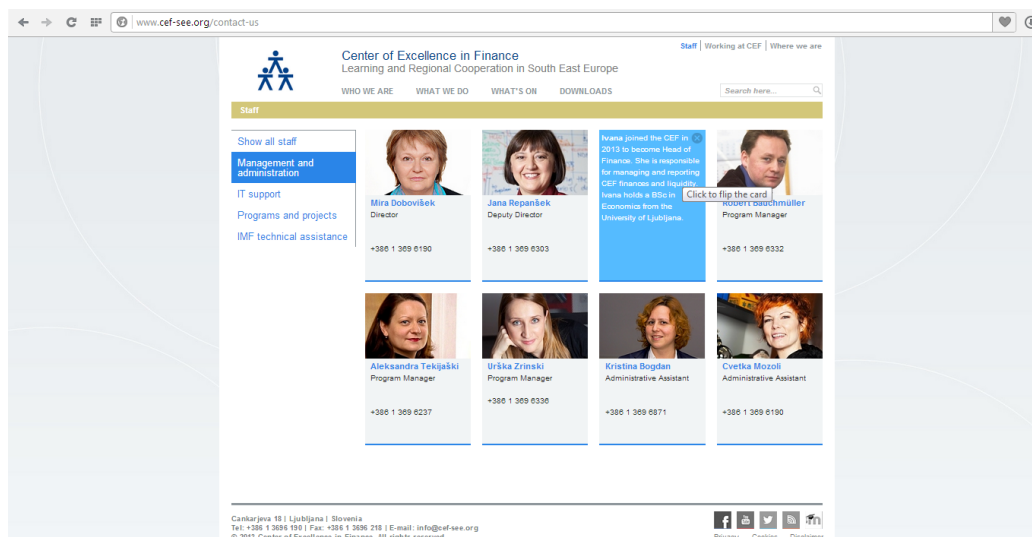
Izsek kode 4.4: Zelo skrajšana in poenostavljena Ext JS koda ogrodja za prikaz kontaktov.

Naša komponenta omogoča iskanje kontaktov, dodajanje novih kontaktov preko vnosne maske, njihovo urejanje in brisanje. Za uporabniški vmesnik sta pomembni dve polji, in sicer polje **order**, ki določa vrstni red kontaktov na strani, ter polje **visible**, ki pove je kontakt sploh prikazan na strani. Spremembo vrstnega reda lahko uredimo tudi z udobnim drag'n'drop v seznamu zalednega vmesnika. Polje **position** je aktualno samo za stran CEF, polje **pozicija** pa samo za stran CRF. Vsako lastno komponento je na koncu razvoja, po posodobitvi oziroma kadarkoli, mogoče zapakirati v poseben transportni paket (Transport Package). To so datoteke tipa transport.zip, s katerimi lahko transportiramo skoraj vse iz podatkovne zbirke

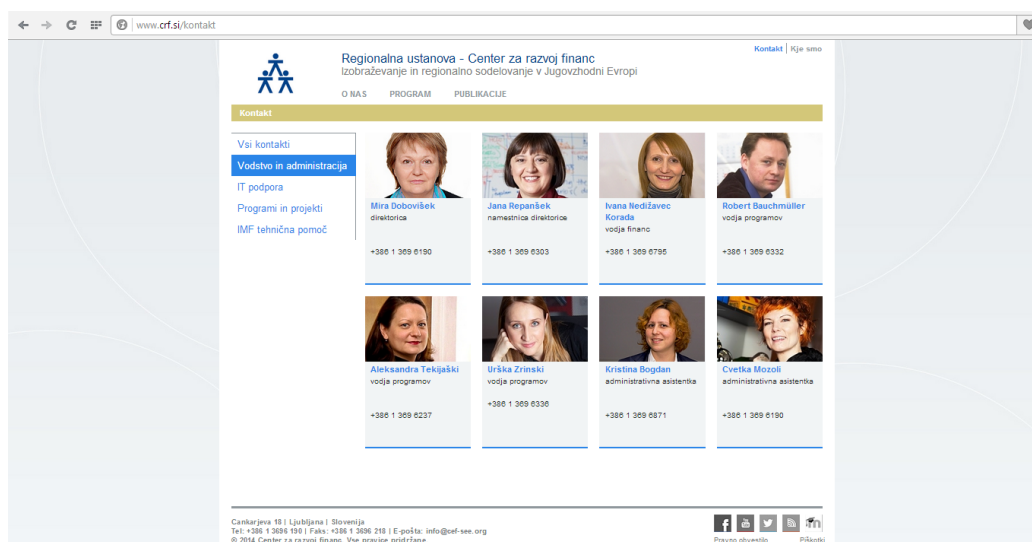


Slika 4.22: Contacts: lastna zaledna komponenta: urejanje iskanega kontakta.

MODx, datotečnega sistema MODx ter skripte za namestitve paketa v drug sistem MODx preko komponente Package Management [74]. To smo vsakokrat storili tudi mi, ko smo iz testnega okolja, kjer smo razvijali ali posodabljali komponento Contacts, želeli prenesti komponento v produkcijsko okolje. Transportni paket `contacts-#. #-release.transport.zip`, kjer `#. #` v imenu predstavlja verzijo, smo preko Package Managementa elegantno transportirali novo ali posodobljeno komponento Contacts. Na enak način so bile ustvarjene vse javne komponente, ki so na voljo na spletnem naslovu MODx Extras [31] ali kar preko komponente Package Management. K prilogi B smo dodali tudi vsebino HTML (Content) strani (Resource) Staff, ki predstavlja stran kontaktov spletnega mesta CEF. Stran Staff ima za Template izbran BaseTemplate, katerega koda je dodana k prilogi A.



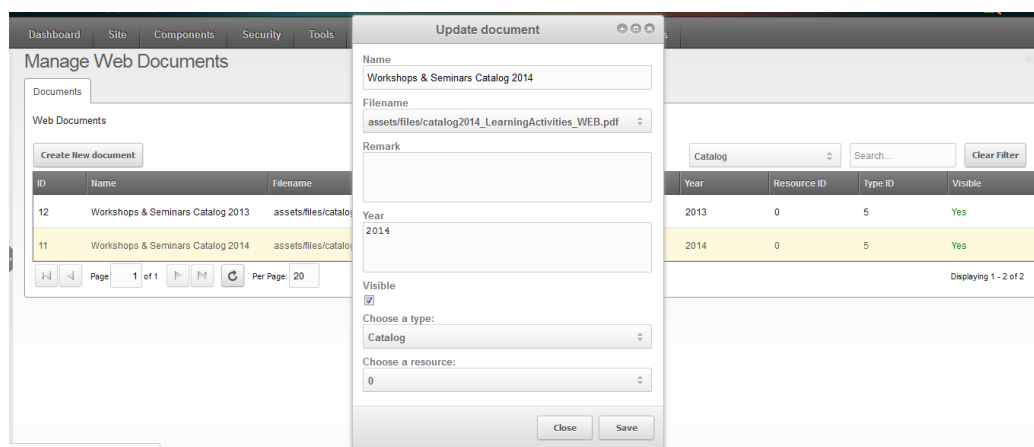
Slika 4.23: Contacts: odzivni uporabniški vmesnik na strani CEF: filter kontaktov preko levega menija in obračanje kartic kontaktov na klik kartice.



Slika 4.24: Contacts: odzivni uporabniški vmesnik na strani CRF: filter kontaktov preko levega menija.

Web Documents (CEF Reports)

To je zaledna komponenta, ki temelji na komponenti Contacts, a je na izgled nekoliko enostavnejša. Omogoča upravljanje z dokumenti spletne strani CEF. Gre predvsem za dokumente več tipov, npr.: letno poročilo, poročilo kratke delavnice, katalog, brošura . . . Tem dokumentom lahko priredimo tudi relacijo z izbranim Resource (stranjo).



Slika 4.25: Web Documents: urejanje dokumenta preko seznama zalednega vmesnika.

Schema vsebuje dva objekta, Reports ter objekt ReportType. Objekt Reports vsebuje še dve relaciji, ena predstavlja povezavo z objektom modResource (Resource), druga pa z objektom ReportType.

```

1 <?xml version="1.0" encoding="UTF-8"?>
2 <model package="cef_reports" baseClass="xPDObject" platform="mysql"
  defaultEngine="MyISAM" version="1.1">
3
4   <object class="Reports" table="reports" extends="xPDObject">
5     <field key="id" dbtype="int" precision="25" attributes="unsigned"
      phptype="integer" null="false" index="pk" generated="native" />
6     <field key="resourceID" dbtype="int" precision="25"
      attributes="unsigned" phptype="integer" null="true" />
7     <field key="year" dbtype="int" precision="10" attributes="unsigned"
      phptype="integer" null="true" />

```

```

8      <field key="name" dbtype="varchar" precision="255" phptype="string"
      null="false" />
9      <field key="filename" dbtype="varchar" precision="255" phptype="string"
      null="false" />
10     <field key="remark" dbtype="varchar" precision="255" phptype="string"
      null="true" />
11     <field key="visible" dbtype="tinyint" precision="1" phptype="integer"
      null="false" />
12     <field key="typeID" dbtype="tinyint" precision="3"
      attributes="unsigned" phptype="integer" null="false" />
13
14     <aggregate alias="Resource" class="modResource" local="resourceID"
      foreign="id" cardinality="one" owner="foreign" />
15     <aggregate alias="Type" class="ReportType" local="typeID" foreign="id"
      cardinality="one" owner="foreign" />
16
17     <index alias="PRIMARY" name="PRIMARY" primary="true" unique="true"
      type="BTREE" >
18       <column key="id" length="" collation="A" null="false" />
19     </index>
20   </object>
21
22   <object class="ReportType" table="reports_type" extends="xPDOObject">
23     <field key="id" dbtype="tinyint" precision="3" attributes="unsigned"
      phptype="integer" null="false" index="pk" generated="native" />
24     <field key="name" dbtype="varchar" precision="45" phptype="string"
      null="false" />
25
26     <index alias="PRIMARY" name="PRIMARY" primary="true" unique="true"
      type="BTREE" >
27       <column key="id" length="" collation="A" null="false" />
28     </index>
29   </object>
30
31 </model>

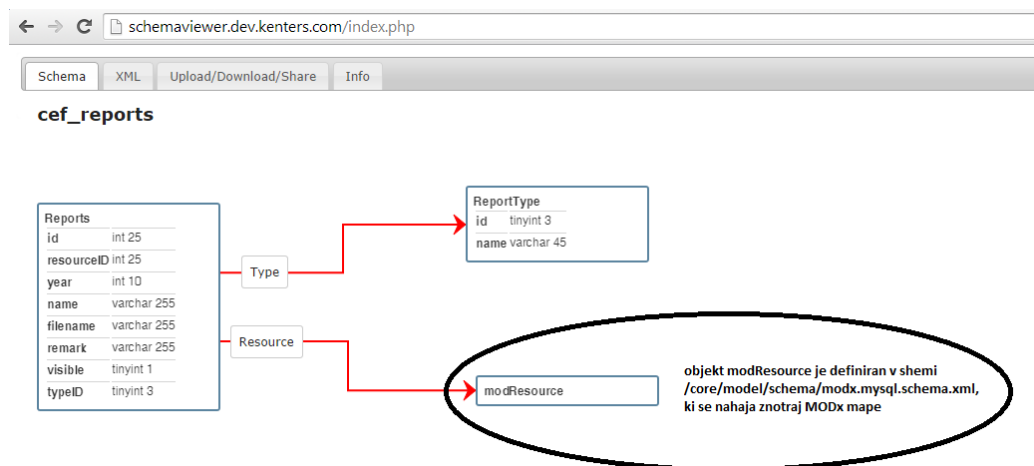
```

Izsek kode 4.5: Shema za Web Documents komponento.

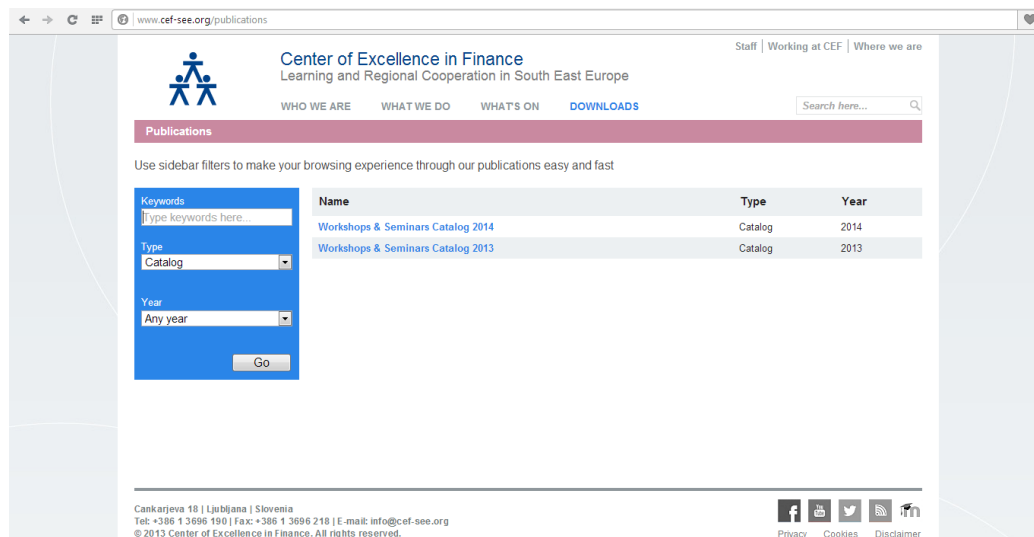
Vizualizacija sheme prikazuje Slika 4.26, zgenerirana s spletnim orodjem xPDO Schema Visualizer [30], kjer rdeča barva pomeni agregatno relacijo (aggregate relation), zelena pa kompozitno (composite relation) [70].

Na uporabniškem vmesniku so ti dokumenti dostopni preko enostavnega iskalnika. Tudi pri tej komponenti smo morali sami razviti Snippet za izgra-

dnjo odzivnega uporabniškega vmesnika s spletnim obrazcem ter prikazom zadetkov na podlagi izbranih filtrov.



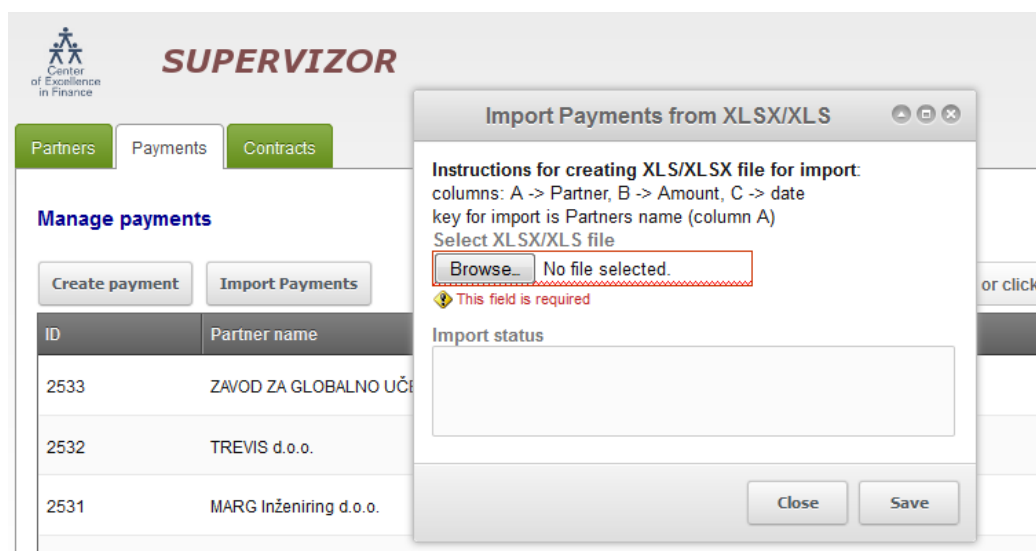
Slika 4.26: Vizualizacija sheme z orodjem xPDO Schema Visualizer za komponento CEF Reports.



Slika 4.27: Web Documents: iskalnik uporabniškega vmesnika in zadetki iskanja preko filtrov.

CEF Supervisor

To je naša zadnja lastna zaledna komponenta, ki je od vseh treh glede vmesnika najbolj posebna. Prvotno je bila razvita za prikaz izplačil proračunskih sredstev za plačilo blaga poslovnim subjektom in drugim pravnim osebam, razen nakazil fizičnim osebam. Komponenta je bila tako prvotno realizirana v dveh jezičkih: prvi je predstavljal tabelo partnerjev, drugi pa tabelo izvedenih izplačil na mesečni ravni. Čez čas se je izkazalo, da bi bilo zelo udobno, če bi obstajala funkcionalnost importa plačil za poslovnim subjektom. To akcijo smo realizirali z razvojem lastnega import procesorja (razširitev razreda `modObjectProcessor`), ki za vhod preko dialog okna prejme XLSX datoteko in nato za vsako vrstico iz tabele kliče create procesor za objekt Payments, ki poskrbi za pravilen vnos posamičnega objekta (vrstice plačila) v podatkovno zbirko [76]. Če poenostavimo, so procesorji PHP koda, ki se izvede ob vsaki akciji zalednega vmesnika komponente za izbrane objekte [75]. Še nekoliko kasneje pa se je pojavila zahteva, da je treba javno objaviti tudi nakazila avtorskih honorarjev in svetovalnih pogodb fizičnih oseb, zato smo za te nove



Slika 4.28: Supervisor: modalno okno s statusno vrstico akcije (za import plačil iz XLSX datoteke v podatkovno zbirko).

potrebe komponento razširili tako, da smo v isti zaledni vmesnik dodali nov jeziček, v katerega smo dodali kar dve tabeli, ena predstavlja pogodbe, druga pa izplačila pogodb. Zaradi lažjega nadzora za prikaz nad vsemi izplačili smo to dvoje združili v skupni komponenti, prikaz in iskalnik na uporabniškem vmesniku pa je ločen, in sicer na dveh straneh spletnega mesta CRF.

```

1 <?xml version="1.0" encoding="UTF-8"?>
2 <model package="supervizor" baseClass="xPDOObject" platform="mysql"
  defaultEngine="MyISAM" version="1.1">
3   <object class="Partners" table="partner" extends="xPDOSimpleObject">
4     <field key="name" dbtype="varchar" precision="150" phptype="string"
      null="false" />
5     <field key="mat" dbtype="int" precision="10" attributes="unsigned"
      phptype="integer" null="false" />
6     <field key="remark" dbtype="varchar" precision="255" phptype="string"
      null="true" />
7     <field key="visible" dbtype="tinyint" precision="1" phptype="integer"
      null="false" />
8
9     <composite alias="Payments" class="PartnerPayments" local="id"
      foreign="partnerid" cardinality="many" owner="local" />
10
11   </object>
12
13   <object class="PartnerContractPayouts" table="partner_contract_payouts"
      extends="xPDOSimpleObject">
14     <field key="contractid" dbtype="int" precision="10" phptype="integer"
      null="false" />
15     <field key="date" dbtype="date" phptype="date" null="false" />
16     <field key="amount" dbtype="decimal" precision="15,2" phptype="float"
      null="false" />
17
18   <aggregate alias="PartnerPayout" class="PartnerContracts"
      local="contractid" foreign="id" cardinality="one" owner="foreign" />
19 </object>
20
21   <object class="PartnerContractTypes" table="partner_contract_types"
      extends="xPDOSimpleObject">
22     <field key="name" dbtype="varchar" precision="150" phptype="string"
      null="false" />
23 </object>
24
25   <object class="PartnerContracts" table="partner_contracts"
      extends="xPDOSimpleObject">

```



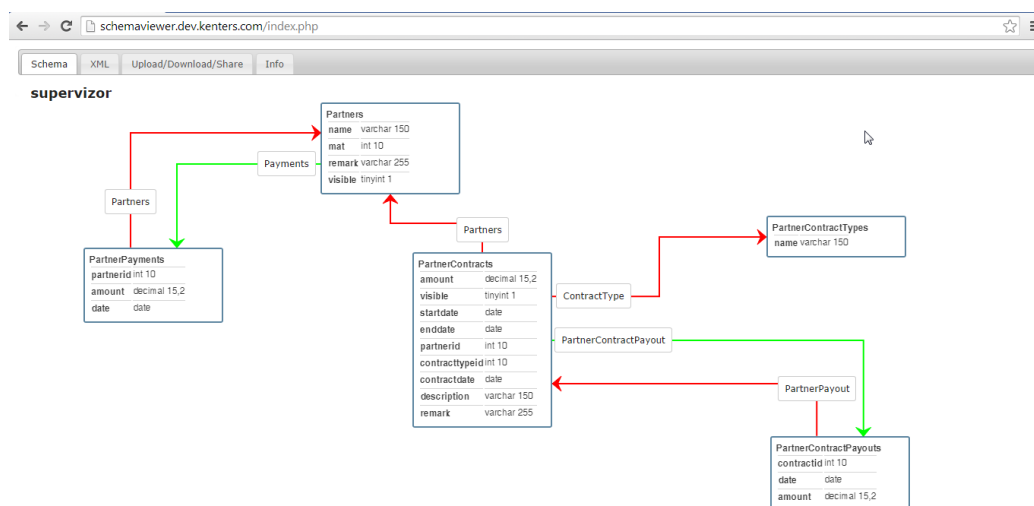
```

26     <field key="amount" dbtype="decimal" precision="15,2" phptype="float"
      null="false" />
27     <field key="visible" dbtype="tinyint" precision="1" phptype="integer"
      null="false" default="0" />
28     <field key="startdate" dbtype="date" phptype="date" null="false" />
29     <field key="enddate" dbtype="date" phptype="date" null="false" />
30     <field key="partnerid" dbtype="int" precision="10" phptype="integer"
      null="false" />
31     <field key="contracttypeid" dbtype="int" precision="10"
      phptype="integer" null="false" />
32     <field key="contractdate" dbtype="date" phptype="date" null="false" />
33     <field key="description" dbtype="varchar" precision="150"
      phptype="string" null="true" />
34     <field key="remark" dbtype="varchar" precision="255" phptype="string"
      null="true" />
35
36     <aggregate alias="Partners" class="Partners" local="partnerid"
      foreign="id" cardinality="one" owner="foreign" />
37     <aggregate alias="ContractType" class="PartnerContractTypes"
      local="contracttypeid" foreign="id" cardinality="one"
      owner="foreign" />
38
39     <composite alias="PartnerContractPayout" class="PartnerContractPayouts"
      local="id" foreign="contractid" cardinality="many" owner="local" />
40
41
42 </object>
43
44 <object class="PartnerPayments" table="partner_payments"
      extends="xPDOSimpleObject">
45     <field key="partnerid" dbtype="int" precision="10"
      attributes="unsigned" phptype="integer" null="false" />
46     <field key="amount" dbtype="decimal" precision="15,2" phptype="float"
      null="false" />
47     <field key="date" dbtype="date" phptype="date" null="false" />
48
49     <aggregate alias="Partners" class="Partners" local="partnerid"
      foreign="id" cardinality="one" owner="foreign" />
50
51 </object>
52 </model>

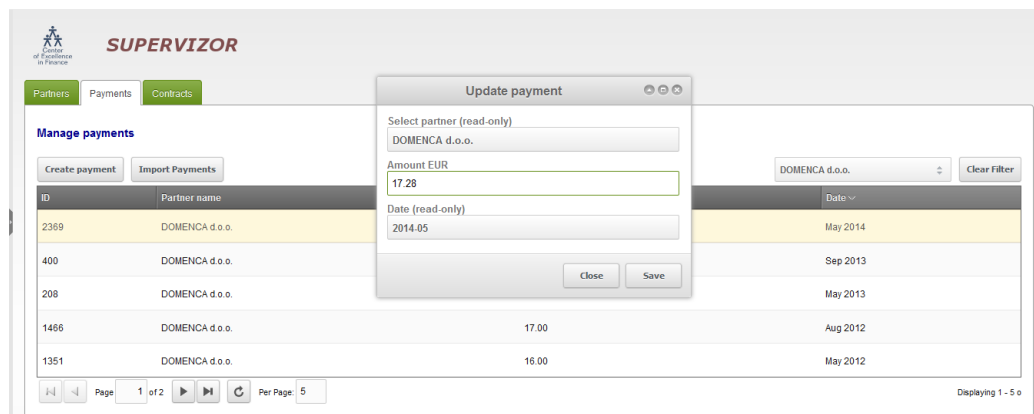
```

Izsek kode 4.6: Shema za Supervizor komponento.

Vizualizacijo sheme prikazuje Slika 4.29, zgenerirana s spletnim orodjem xPDO Schema Visualizer [30], kjer rdeča barva pomeni agregatno relacijo (aggregate relation), zelena pa kompozitno (composite relation) [70].



Slika 4.29: Vizualizacija sheme z orodjem xPDO Schema Visualizer za komponento Supervisor.



Slika 4.30: Supervisor: urejanje plačila blaga ali storitve pravne osebe.

Manage Contracts and add Payouts (right click)

Create contract Add payout

ID	Partner name	Amount EUR
13	Sintegra lab d.o.o.	9.455.00
10	Eudaimonia.pt	4.200.00
15	ODVETNIŠKA DRUŽBA BURAGIC - SOTLAR o.p. ...	4.550.00

Page 1 of 1 Per Page: 10

Filter and edit Payouts

Contract ID	Partner	Amount EUR	Date	ID
11	Zdenka Vidovič	1.000.00	2014-08-20	15
18	Hristina Trajkovska	750.00	2014-08-20	14
4	Thomas Withemson	250.00	2014-07-03	13
5	Bojan Paučovič	750.00	2014-07-03	12
2	André Lundvall	250.00	2014-07-03	11

Page 1 of 3 Per Page: 5

Displaying 1 - 3 of 3

Slika 4.31: Supervizor: urejanje plačila za svetovalno pogodbo.

← → ↻ 🔍 www.crf.si/supervizor

Regionalna ustanova - Center za razvoj financ
Izobraževanje in regionalno sodelovanje v Jugovzhodni Evropi

Kontakt | Kje smo

O NAS PROGRAM PUBLIKACIJE

Supervizor

Uporabite filtre za hitrejšo iskanje

Partner: DOMENCA d.o.o.

Leto: vsa leta

Mesec: vsi meseci

Isči

Partner	Mesec	Vsota v EUR
DOMENCA d.o.o.	maj 2011	16,00
DOMENCA d.o.o.	maj 2012	16,00
DOMENCA d.o.o.	avg 2012	17,00
DOMENCA d.o.o.	maj 2013	16,00
DOMENCA d.o.o.	sep 2013	19,33
DOMENCA d.o.o.	maj 2014	17,28

Podatki so prikazani na podoben način kot jih prikazuje Supervizor, aplikacija Komisije za preprečevanje korupcije. Vir podatkov so poslovne knjige Regionalne ustanove – Centra za razvoj financ (Center), analitična evidenca dobaviteljev.

Slika 4.32: Supervizor: iskalnik uporabniškega vmesnika in zadetki iskanja preko filtrov za plačilo blaga ali storitev pravnih oseb.



Slika 4.33: Supervisor: iskalnik uporabniškega vmesnika in zadetki iskanja preko filtrov za plačila fizičnim osebam.

4.3.9 Lastni Snippet

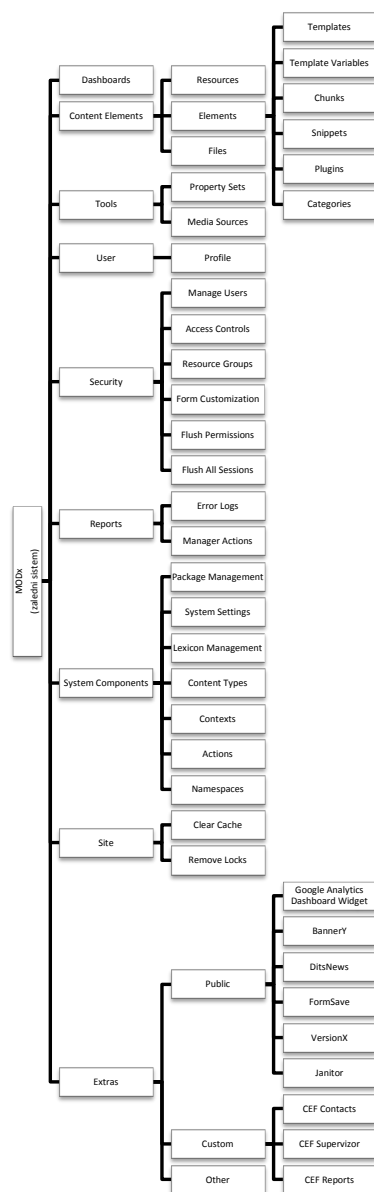
Za doseg želenih funkcionalnosti smo poleg komponent morali razviti tudi nekaj lastnih Snippetov. Napisali smo jih okoli 100, a našteji bomo le nekaj najpomembnejših. Njihove klice lahko najdemo v Template ali katerikoli glavnem elementu.

ime	opis
fbWallAjax	fbWallAjax je Snippet, ki vrača zadnje objave s CEF Facebook zida.
tabsLogic	Za izgradnjo jezičkov na uporabniškem vmesniku, ki predstavlja stranski (levi) meni.
BreadCrumb	Za izgradnjo "drobtin", ki so nekoliko posebne.
betterWayFinder	Za izgradnjo glavnega menija, saj je ta tudi nekoliko poseben.
combineStyleFiles	Združi in minimizira več datotek tipa css in less v eno datoteko css.

ime	opis
<code>addSpecialJS-CSS</code>	Snippet v Templateu, ki poskrbi za posebne primere; doda se JavaScript, CSS ali pa se izvede PHP koda.
<code>ajax</code>	Izvede se ob vsakem ajax klicu na uporabniškem vmesniku.
<code>backgroundImageStyle</code>	Snippet, ki na podlagi slike in parametrov vrne niz za HTML atribut style.
<code>contentTemplate</code>	Vrne vsebino Chunka, ki ima vlogo mini-templatea, ali pa vrne vsebino (Content) strani.
<code>createThumb</code>	Pomožni Snippet, ki na podlagi posebnih parametrov kliče Snippet <code>phpthumbof</code> .
<code>cssSprites</code>	Pomožni Snippet, ki zgnerira CSS Sprite, CSS in HTML na podlagi vhodnih ikon/slik.
<code>dateDuration</code>	Vrne 'date range' niz na podlagi <code>startDate</code> in <code>endDate</code> TV.
<code>do.stuff.if.admin</code>	Snippet v Template, ki se izvede, če smo prijavljeni kot administratorji na uporabniškem vmesniku.
<code>Editikona</code>	Izpiše 'edit ikono', če so izpolnjeni kriteriji za to.
<code>getMyContacts,</code> <code>getMyContracts,</code> <code>getMyPayments,</code> <code>getMyReports</code>	Serijsko posebnih Snippetov, ki vračajo dinamično vsebino za posebne strani.
<code>labPlaceholder,</code> <code>labPlaceholderChunk,</code> <code>labJS-CSS</code>	Serijsko posebnih Snippetov za dinamično nalaganje CCS datotek in generiranje dinamične JavaScript kode, ki se na koncu tudi minimizira. Uporabijo se lahko v katerikoli glavnem elementu.

ime	opis
LoginNew	Izpiše 'adminPanel', če smo prijavljeni preko uporabniškega vmesnika.
myCache	Poseben Snippet za vračanje posebnega predpomnilnika (ali pa se ustvari sam in ga vrne).
newsletterGenerator	Snippet za izgradnjo Newsletter strani (Resource), ki se potem uporabi za pošiljanje Newsletter e-pošte.
prevNext	Snippet, ki zgenerira link za naprej in nazaj pri novicah in dogodkih na uporabniškem vmesniku.
...	In še mnogi drugi.

4.3.10 Struktura MODx



Slika 4.34: Struktura zalednega sistema MODx.

4.4 Namestitev sistema

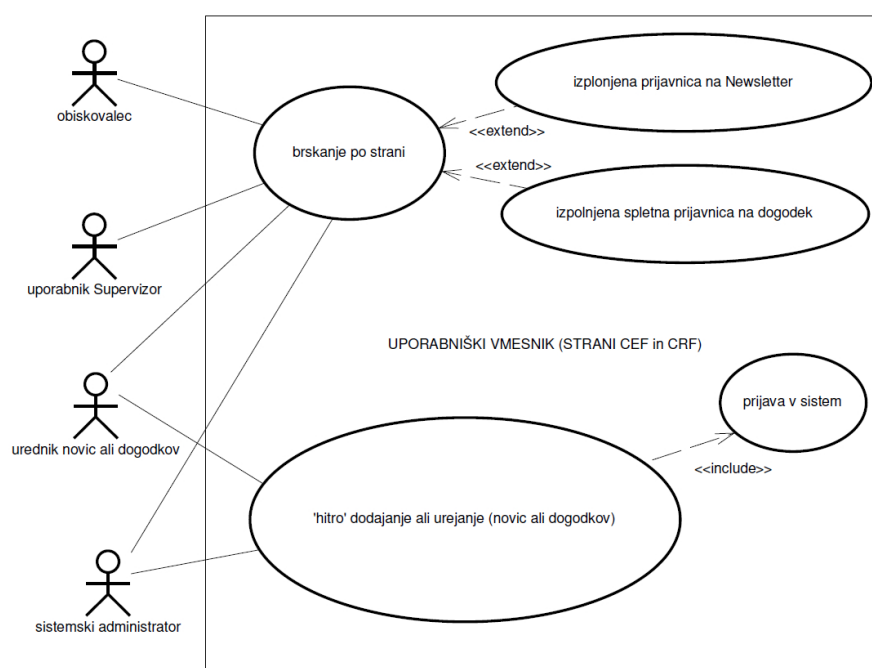
Po analizi in načrtovanju smo prišli do faze namestitve testnega sistema. Namesto primernega strežnika smo uporabili kar običajen PC računalnik, ki je dovolj dober za testni razvoj. Na računalnik smo namestili operacijski sistem Linux Ubuntu, spletni strežnik Nginx, podatkovno bazo MySQL, prevajalnik PHP in procesni upravljalnik php-fpm [34]. Ustrezno smo skonfigurirali vsa omenjena orodja in nato namestili še sistem MODx [35], nato pa smo lahko začeli z razvojem v MODx-u. Po končanem razvoju v testnem okolju smo celoten sistem dobro stestirali in ko smo bili zadovoljni, smo namestili produkcijsko okolje z enakimi orodji, vključno z enako verzijo MODx sistema. Poskrbeli smo še za varnost in vse ostale potrebne mrežne nastavitve. Prenos razvitega sistema iz testnega okolja ni bil težaven, saj smo prenesli celotno MySQL bazo MODx-a ter datotečni sistem MODx-a. Nato smo ponovno pognali namestitev MODx-a (Upgrade Existing Install) zaradi razlik v konfiguraciji ter prilagodili še sistemske nastavitve MODx-a (System Settings). Opravili smo še celovito testiranje sistema. Po tem je bil produkcijski sistem pripravljen za uporabo. V primeru razširitve komponente smo to najprej nadgradili na testnem okolju. Novo različico komponente smo nato iz testnega okolja s transportno skripto transportirali v transportni paket. V produkciji pa smo komponento posodobili z dobljenim transportnim paketom preko Package Managementa.

Poglavje 5

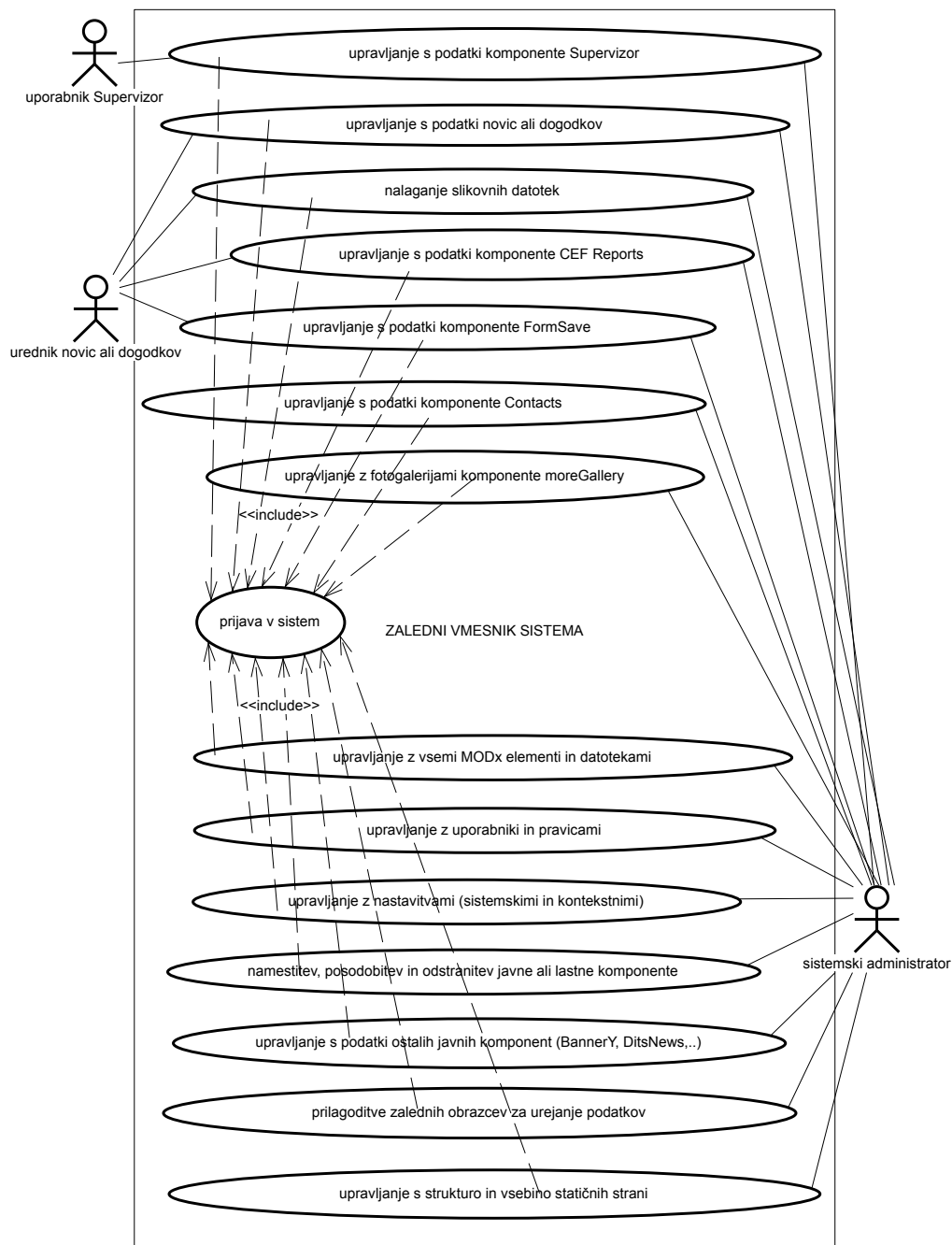
Uporaba sistema in administracija

5.1 Diagram primerov uporabe

Diagrami primerov uporabe (Use Case Diagrams) predstavljajo povezavo med akterji - uporabniki sistema in procesi, ki v sistemu tečejo. Primeri uporabe so običajno namenjeni zajemanju in predstavitvi uporabniških zahtev [78]. S pomočjo diagramov primerov uporabe modeliramo vplive okolja na sistem. Sistem modeliramo kot množico procesov, ki uporabniku ponujajo usluge ali storitve sistema. Hkrati okolje sistema modeliramo kot množico akterjev, ki vplivajo na sistem oziroma od sistema zahtevajo merljiv rezultat [78]. V nadaljevanu sta prikazana dva diagrama primerov uporabe, eden za uporabniški vmesnik (Slika 5.1), drugi pa za zaledni vmesnik (Slika 5.2).



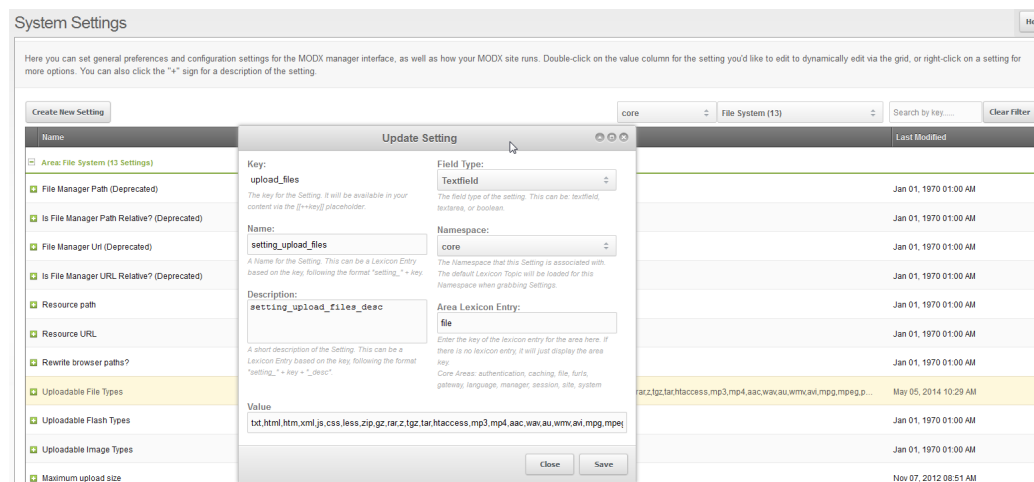
Slika 5.1: Diagram primerov uporabe uporabniškega vmesnika.



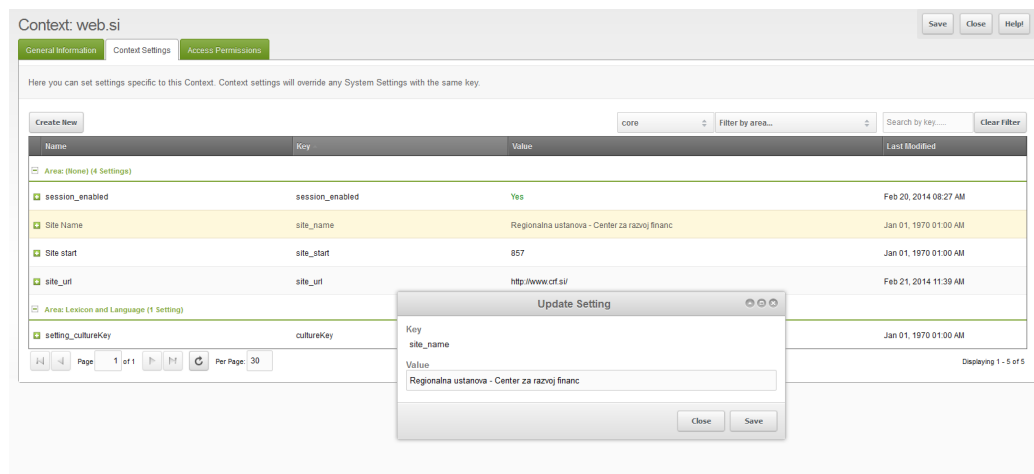
Slika 5.2: Diagram primerov uporabe zalednega vmesnika.

5.2 Skrbniški del sistema

Administratorjem je v zalednem vmesniku omogočeno urejanje raznih nastavitvev kar preko akcij glavnega menija (Slika 5.13).



Slika 5.3: Urejanje sistemskih nastavitev znotraj zalednega vmesnika.



Slika 5.4: Urejanje nastavitev za določen kontekst znotraj zalednega vmesnika

5.2.1 Uporabniki in pravice

MODx ima zelo močan nabor akcij za upravljanje z uporabniki, pravicami uporabnikov in skupinami uporabnikov. Preko teh akcij lahko prilagajamo zaledni vmesnik poljubnih uporabnikov.

Center of Excellence in Finance
MODx Revolution 2.2.14-pl (advanced)

Dashboard Site Components **Security** Tools Reports System User Support CEF Members

Resources Elements Files

Search...

web

- Home (1)
- Menu-Top (357)
- Menu-Header (476)
 - Who We Are (477)
 - About us (481)
 - Our members (482)
 - Our partners (483)
 - Our governance (496)
 - What We do (478)
 - What's on (728)
 - Downloads (480)
 - Menu-Bottom (814)
 - Events (8)

Manage Users
Add, update, and assign permissions to users.

Access Controls
Manage user groups, roles, and access policies.

Resource Groups
Manage the groups resources belong to.

Form Customization
Customize manager forms by security permissions.

Flush Permissions
Flush all permissions and reload the cache.

Flush All Sessions
Flush all sessions and logout all users.

Center of Excellence in Finance

800
600
400
200
0

2014-09-09 2014-09-10

DATE	VISITS	ABSOLUTE UNIQUE VISITORS
2014-09-09	171	148
2014-09-10	151	118
2014-09-11	133	110
2014-09-12	190	138

Slika 5.5: Security nastavitve znotraj zalednega vmesnika.

Access Controls

User Groups Roles Access Policies Policy Templates

Manage user memberships in groups and apply roles.

New User Group

User Groups

- anonymous
- Administrator (1)
- Members (2)
- Subscribers (3)
- Supervisor (4)

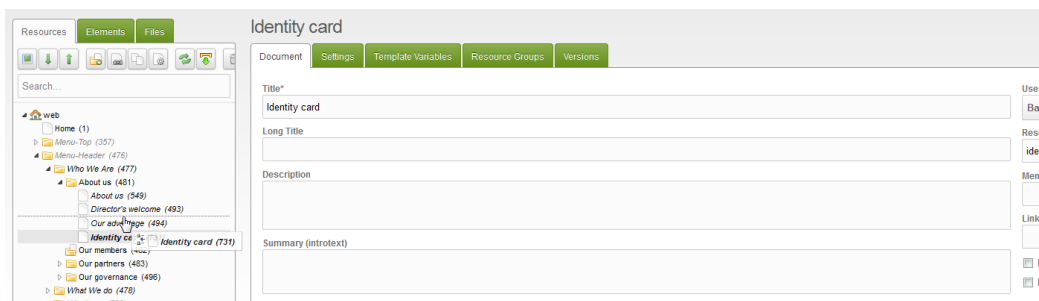
Slika 5.6: Upravljanje s pravicami za skupine uporabnikov

5.2.2 Urejanje opisa strani

Glavno vsebinsko polje za strani je polje vsebina (Content), sicer pa vsebino strani lahko predstavljajo še drugi ločeni gradniki, ki jih nato sestavimo v celoto. Vsebino lahko urejamo.

Opis strani je lahko razdeljen v ločena polja, v kolikor gre za posebne vsebine, sicer pa imajo strani glavno vsebinsko polje *vsebina* (Content), ki se lahko ureja preko uporabniškega ali zalednega urejevalnika vsebine:

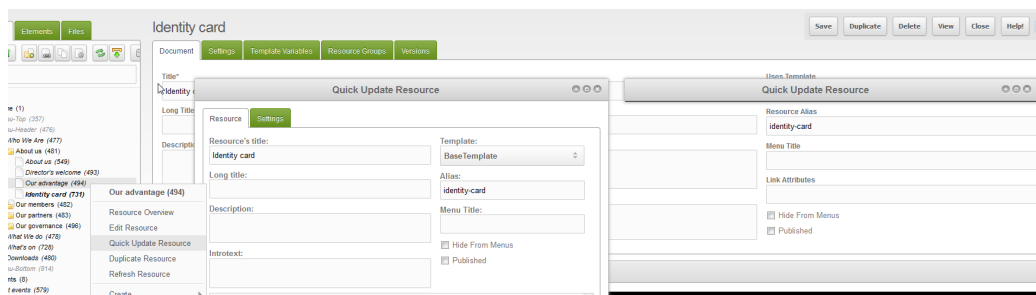
- za glavne dinamične vsebine hitro preko uporabniškega vmesnika za urejanje vsebin (Slika 5.18);
- vse vsebine ter objekti, ki lahko predstavljajo kodo, so vedno na voljo v zalednem vmesniku, če so za to dodeljene ustrezne uporabniške pravice.



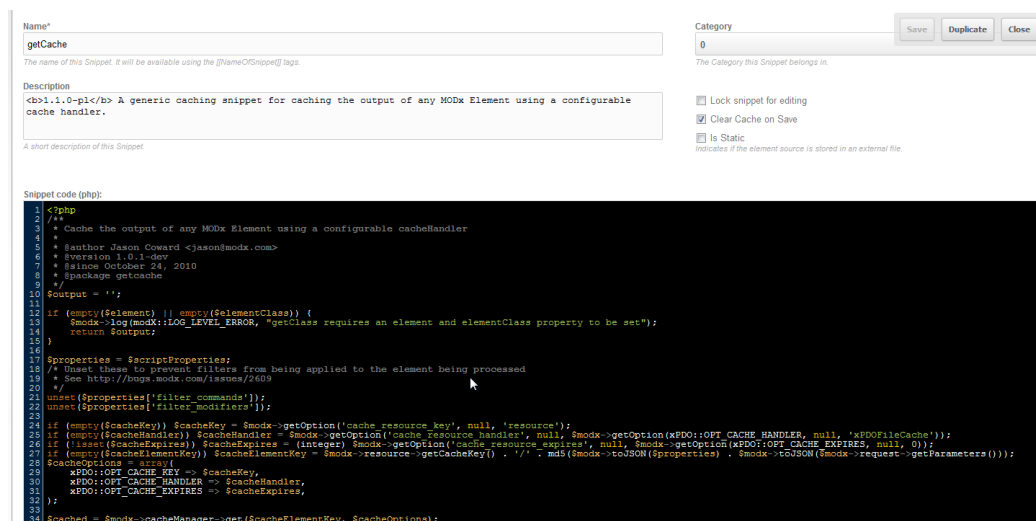
Slika 5.7: Primer urejanja drevesne strukture glavnega menija in podmenijev na uporabniškem vmesniku.

Vsebino strani je mogoče urejati s klikom na Resource v normalnem načinu ya urejanje strani, lahko pa tudi preko hitrega načina v pomožnih oknih.

Za administratorje je zelo priljubljeno urejanje kode v urejevalniku s podporo kodnega obarvanja.



Slika 5.8: Primer urejanja vsebine poljubne strani v zalednem vmesniku.

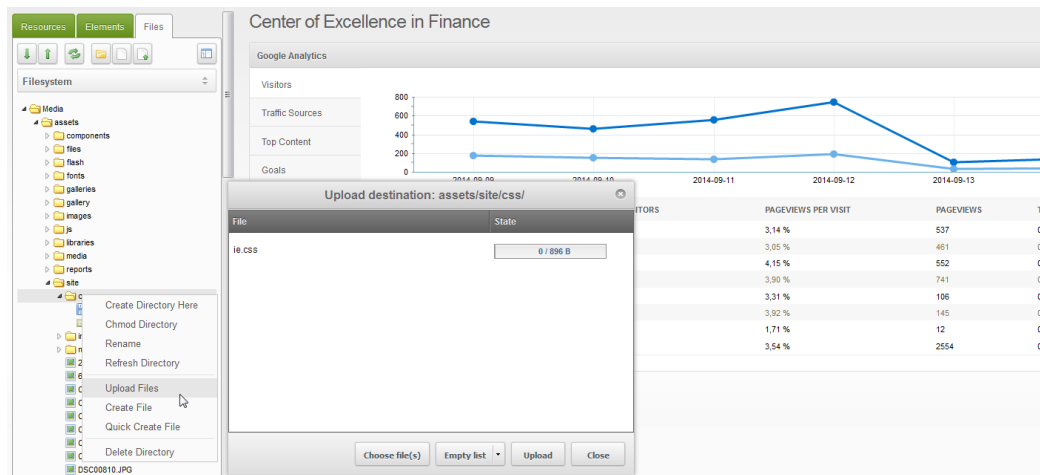


Slika 5.9: Urejanje kode za administratorje v zalednem vmesniku.

5.2.3 Datoteke CSS

Datoteke CSS se nahajajo znotraj mape /assets in jih lahko naložimo kar preko zalednega vmesnika (jeziček Files).

Za doseg hitrejšega nalaganja strani smo razvili Snippet `combineStyleFiles`, ki omogoča, da kot vhodni parameter files prejme seznam CSS ali LESS datotek, ki jih združi v eno datoteko in jo nato še minimizira ter vrne pot nove skupne CSS datoteke.



Slika 5.10: Primer nalaganja CSS datoteke v datotečni sistem MODx.

```

1  [[!myCache?
2    &snippet='combineStyleFiles'
3    &myCacheFolder='mycss'
4    &key='myCSS'
5
6    &targetPath='assets/site/'
7    &targetFilename='screen.v6.css'
8
9    &files='assets/templates/foundation4/css_new/normalize.min.css,
10           assets/templates/foundation4/css_new/foundation.min.css,
11           assets/site/simple.css,
12           assets/site/css/styles.less,
13           assets/js/meanmenu/meanmenu.css'
14 ]]

```

Izsek kode 5.1: Konkreten primer uporabe Snippet combineStyleFiles v kombinaciji myCache Snippet znotraj Template.

Za uporabo CSS datoteke lahko na posebnih straneh uporabimo tudi svoj Snippet labPlaceholder preko parametra css, kot to prikazuje Izsek kode 5.2.

5.2.4 Datoteke JavaScript

Datoteke JS se prav tako nahajajo znotraj mape /assets in jih lahko naložimo kar preko zalednega vmesnika (jeziček Files), kot prikazuje Slika 5.10 za nalaganje CSS. JavaScript datoteke in objekte Chunk z inline JavaScript pa lahko uporabimo z lastnim Snippetom labPlaceholder.

```
1 [[!labPlaceholder?  
2     &js='assets/js/vectormap/jquery-jvectormap-1.2.2.min.js,  
3         assets/js/vectormap/jquery-jvectormap-europe-mill-en.js'  
4  
5     &css='assets/js/vectormap/jquery-jvectormap-1.2.2.css'  
6  
7     &inline='var targetCode = '[+code]'; if (targetCode=='') {  
8         $('#introOurMembers').show(); } '  
9  
10    &chunk='countriesShow'  
11 ]]
```

Izsek kode 5.2: Konkreten primer uporabe lastnega labPlaceholder Snippeteta znotraj vsebine posebne strani.

5.2.5 Slike

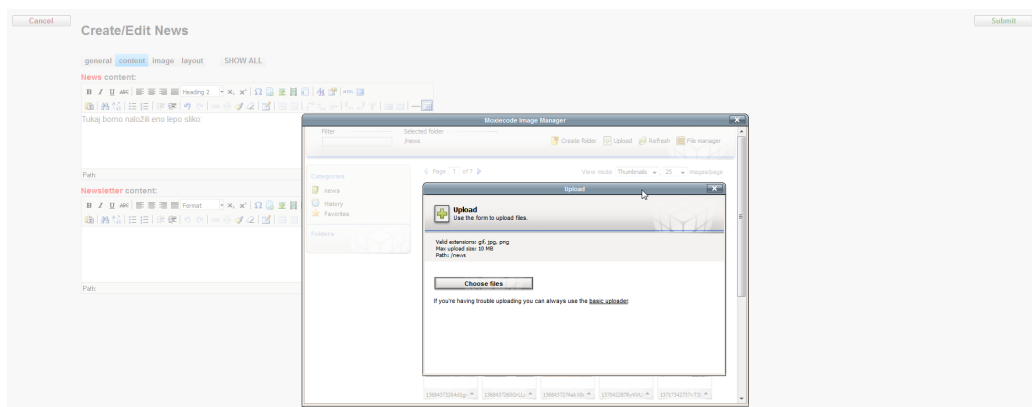
Nalaganje slik se lahko opravi enako, kot to lahko počnemo za vse druge tipe datotek v mapo assets, kot to prikazuje Slika 5.10.

Uredniki in administratorji lahko nalagajo slike tudi preko uporabniškega vmesnika, ko urejajo novico (Slika 5.11).

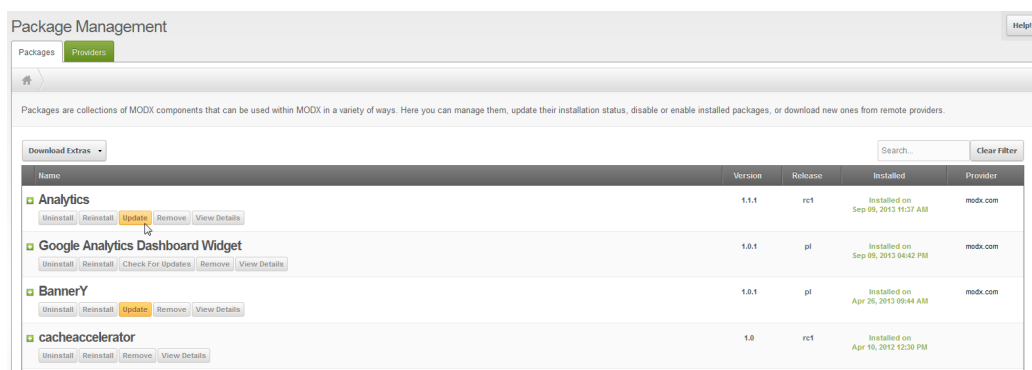
Za manipulacijo slik je uporaben Snippet phpThumb, ki ga lahko uporabimo znotraj drugih Snippetov ali kar v Template ali vsebini strani.

5.2.6 Komponente

S komponentami lahko enostavno upravljamo preko Package Management menija znotraj zalednega vmesnika, kjer lahko namestimo nove komponente, jih posodabljammo ali tudi odstranimo.



Slika 5.11: Primer nalaganja slik preko uporabniškega vmesnika.



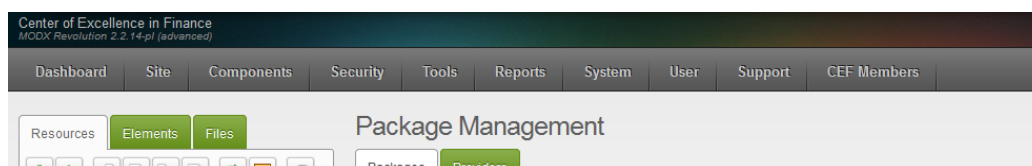
Slika 5.12: Uporaba Package Managementa komponente, vgrajene v MODx

5.2.7 Ostale akcije glavnega menija

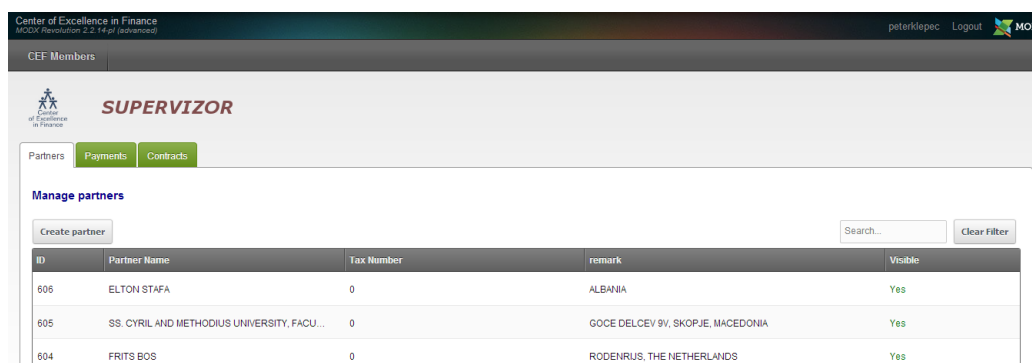
Glavni meni vsebuje veliko uporabnih akcij, do katerih lahko dostopa administrator.

Podmenije in akcije glavnega menija v zalednem vmesniku lahko tudi poljubno spreminjamo za vsako skupino uporabnikov. Prav tako jih lahko npr. z uporabo Pluginov preusmerimo na določeno komponento po uspešni zaledni prijavi za določeno skupino uporabnikov.

5.3. UREDNIŠKI DEL SISTEMA ZA UPRAVLJANJE Z VSEBINAMI 71



Slika 5.13: Glavni meni zalednega vmesnika za administratorje



Slika 5.14: Zelo omejen glavni meni oziroma celotni zaledni vmesnik posebne skupine uporabnikov, ki uporabljajo samo Supervisor komponento.

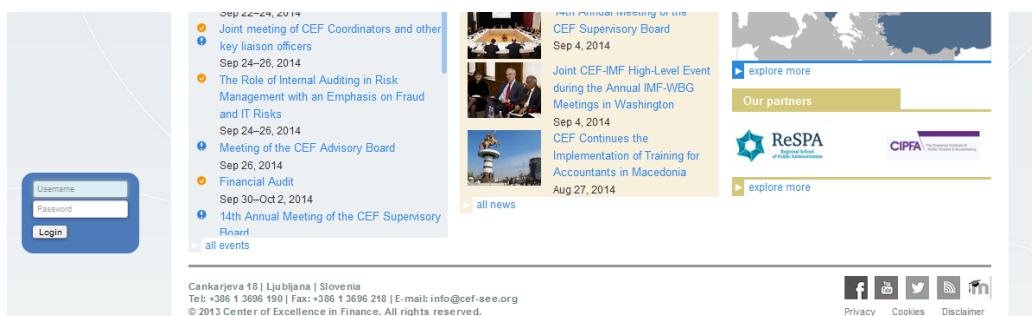
5.3 Uredniški del sistema za upravljanje z vsebinami

5.3.1 Prijava preko uporabniškega vmesnika

V sistem se lahko prijavimo preko prijave uporabniškega vmesnika, če smo kot uporabnik v ustrezni skupini. Za prikaz prijavnega okna uporabniškega vmesnika je potrebna posebna akcija uporabnika, za katero vedo le uporabniki sistema.

Z uspešno tako prijavo dobimo možnost urejanja le najvažnejših dinamičnih vsebin, to so novice in dogodki.

Takšno urejanje je zelo udobno za manj zahtevne uporabnike in je tudi zelo hitro. Urejanje preko uporabniškega vmesnika predstavlja razširitev funkcionalnosti, kar je bila tudi ena od naših zahtev v procesu analize.



Slika 5.15: prijava v sistem preko uporabniškega vmesnika



Slika 5.16: Vstopna stran CEF: vidnih več vsebin (npr. “oranžni dogodki”, ki še niso javni) in ikone za urejanje dinamičnih vsebin, na voljo samo za prijavljene.



Slika 5.17: Stran dogodka: “hitra” ikona za urejanje (zgoraj levo v admin plošči), na voljo samo za prijavljene.

Create/Edit Events

general content layout media SHOW ALL

Published ☒

Title:
Financial Reporting on General Government and its Sub-sectors

the location (for empty location please type single space):
Ljubljana, Slovenia

topic
-

special topic
-

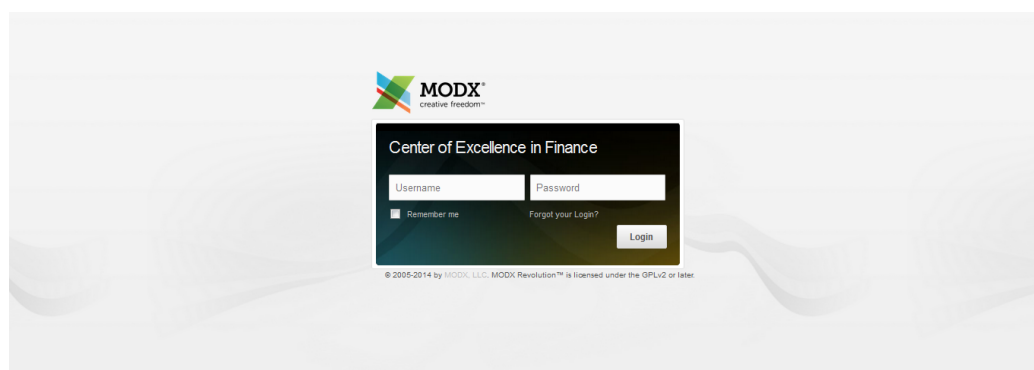
Start Date:
2014-09-22

Cancel Submit

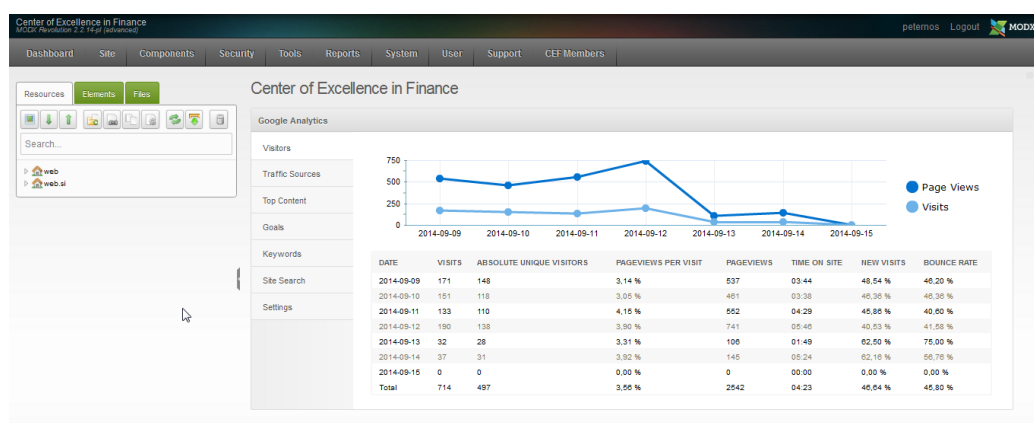
Slika 5.18: Urejanje preko uporabniškega vmesnika: preprost vmesnik, a hiter in učinkovit.

5.3.2 Zaledna prijava

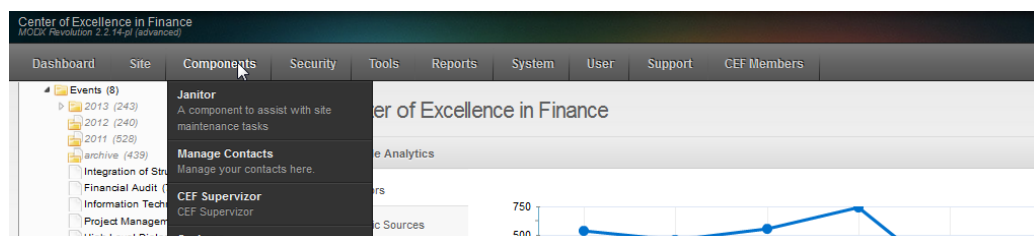
V sistem se lahko prijavimo preko zaledne prijave, če imamo za to dodeljene pravice, za URL naslov prav tako vedo samo uporabniki. MODx v osnovi podpira samo zaledno prijavo. Običajno lahko uporabniki dostopajo do objektov strani (Resource), datotečnega sistema za nalaganje datotek in preko menija Components do dodatkov ali komponent (Extra), ki omogočajo urejanje posebnih vsebin (razširitev). Administratorji pa imajo možnost urejanja praktično vsega, torej poleg vse vsebine tudi programske kode, ki poskrbi za dinamično prikazovanje vsebine na strani, ali pa do komponente za upravljanje z uporabniki etc.



Slika 5.19: Zaledna prijava v sistem.



Slika 5.20: Prikaz nadzorne plošče (Dashboard) takoj po zaledni prijavi.



Slika 5.21: Meni komponent (Components).

Poglavje 6

Sklepne ugotovitve

Odločitev za sistem MODx je bila zagotovo prava odločitev. Z njim smo dosegli vse cilje, ki smo si jih zastavili, ter realizirali še kaj dodatnega. Zadovoljni so tudi vsi akterji, ki sistem uporabljajo. Prepričani smo, da nam bo sistem dobro služil in se ne bojimo novih izzivov na področju spletne predstavitve. MODx je bila prava izbira tudi zato, ker se nam ni tako zelo mudilo z rezultatom in smo si lahko privoščili več časa še za učenje in spoznavanje sistema.

6.1 Ovire

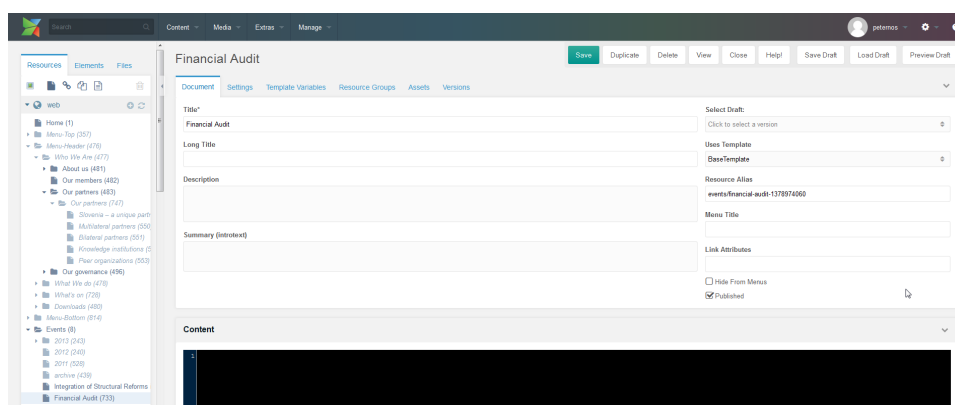
Težav s sistemom MODx nismo imeli, zato smo na MODx lahko ponosni. Smo pa pri razvoju naleteli na nekaj ovir, ki smo jih nato uspešno rešili. To so:

- vsebina in struktura strani (ta se je ponekod razvijala inkrementalno, včasih pa se je poleg vsebine spremenil tudi del strukture strani);
- izbira fonta (spremenili kar nekaj fontov, preden smo se dokončno odločili za pravega);
- iskanje uporabnih in delujočih javnih JavaScript knjižnic (dostikrat je bila potrebna modifikacija knjižnice, največkrat pa smo funkcionalnost potem razvili kar sami);

- podpora za brskalnik Internet Explorer 8 (ponekod je to pomenilo skoraj dvojno delo pri programiranju z JavaScript in CSS, npr. slider in interaktivna mapa na prvi strani);
- design strani smo prejeli v obliki PDF, kar je za nas, razvijalce, pomenilo, da smo morali razviti CSS sami (v ceno designa nismo dobili kode CSS);
- včasih smo se težko odločili za izgled določenih delov strani, saj smo bili različnih mnenj;
- če bi imeli na voljo manj časa, verjetno z MODx ne bi pravočasno prišli na cilj, saj ga je potrebno dobro spoznati (še posebno Ext JS, če želimo imeti odličen zaledni vmesnik).

6.2 Predlogi izboljšav

- Posodobitev MODx-a na zadnjo različico, ki ima sedaj povsem prenovljeni zaledni vmesnik z novo temo (posodobitev je že bila opravljena na testnem strežniku s pozitivnim odzivom in v kratkem bomo to opravili tudi v produkciji);



Slika 6.1: Testno okolje: nova različica MODx-a, testiramo tudi nov lasten Plugin, ki doda novo funkcionalnost: Save Draft, Load Draft, Preview Draft (v vmesniku gumbi zgoraj).

- prevedba nekaterih programskih kod v zaledno komponento za udobnejše urejanje;
- razvoj lastne komponente za preprosto izdelavo lastnih komponent in export v transport package (to trenutno manjka MODx-u).

6.3 Prednosti

MODx ima izreden zaledni vmesnik, ki ponuja razvoj lastnih komponent po meri z uporabo obstoječega Ext JS ogrodja (lahko bi uporabili poljubno knjižnico), sicer pa so prednosti naštetje že v Uvodu.

Slike

2.1	Resources v zalednem vmesniku MODx.	7
2.2	Glavni elementi v zalednem vmesniku MODx.	8
2.3	Priporočena prepletanja elementov znotraj Template [21].	9
2.4	Splošna polja strani.	10
2.5	Nastavitvena polja strani.	11
4.1	Seznam kontekstov v zalednem vmesniku.	21
4.2	xPDO Schema Visualizer: model za shemo zaledne komponente BannerY, ki jo uporabljamo v našem sistemu.	22
4.3	Template in VSEBINA strani.	24
4.4	Spletna prijavnica za dogodke na strani CEF.	25
4.5	Urejanje Template Variables preko zalednega vmesnika za izbran dogodek.	26
4.6	Obrezovanja slike preko uporabniškega vmesnika za izbrano novico.	27
4.7	Uporabniški vmesnik galerije.	28
4.8	Seznam institucij v zaledni komponenti BannerY.	29
4.9	Urejanje posamezne institucije v zaledni komponenti BannerY preko seznama.	30
4.10	Seznam poslanih in neposlanih Newsletterjev v zaledni komponenti DitsNews.	31
4.11	Urejanje naročnika v zaledni komponenti DitsNews preko seznama.	31

4.12 Seznam prijav različnih spletnih prijav v zaledni komponenti FormSave.	32
4.13 Če je Plugin VersionX aktiven, se pri urejanju preko zalednega vmesnika pojavi jeziček Versions, preko katerega se lahko opravi povrnitev.	33
4.14 VersionX: celoten nadzor nad vsemi različicami glavnih vse- binskih objektov in možnost povrnitve, dostopno preko menija.	33
4.15 Google Analytics Dashboard Widget: prikaz statistike Google Analytics po zaledni prijavi.	34
4.16 Janitor: odličen pri iskanju po vsebinskih gradnikih znotraj MODx Revolution.	34
4.17 MoreGallery: enostavno nalaganje slik, reorganizacija slik, možnost opisa za vsako sliko.	35
4.18 Razvoj lastnih komponent: pomoč na vsakem koraku.	39
4.19 Razvoj modela: smer razvoja je poljubna [69].	40
4.20 Vizualizacija sheme z orodjem xPDO Schema Visualizer za komponento Contacts.	41
4.21 Contacts: lastna zaledna komponenta: za udobno upravljanje s kontakti za obe spletni mesti.	44
4.22 Contacts: lastna zaledna komponenta: urejanje iskanega kon- takta.	46
4.23 Contacts: odzivni uporabniški vmesnik na strani CEF: filter kontaktov preko levega menija in obračanje kartic kontaktov na klik kartice.	47
4.24 Contacts: odzivni uporabniški vmesnik na strani CRF: filter kontaktov preko levega menija.	47
4.25 Web Documents: urejanje dokumenta preko seznama zale- dnega vmesnika.	48
4.26 Vizualizacija sheme z orodjem xPDO Schema Visualizer za komponento CEF Reports.	50

4.27 Web Documents: iskalnik uporabniškega vmesnika in zadetki iskanja preko filtrov.	50
4.28 Supervizor: modalno okno s statusno vrstico akcije (za import plačil iz XLSX datoteke v podatkovno zbirko).	51
4.29 Vizualizacija sheme z orodjem xPDO Schema Visualizer za komponento Supervizor.	54
4.30 Supervizor: urejanje plačila blaga ali storitve pravne osebe. . .	54
4.31 Supervizor: urejanje plačila za svetovalno pogodbo.	55
4.32 Supervizor: iskalnik uporabniškega vmesnika in zadetki iskanja preko filtrov za plačilo blaga ali storitev pravnih oseb. . . .	55
4.33 Supervizor: iskalnik uporabniškega vmesnika in zadetki iskanja preko filtrov za plačila fizičnim osebam.	56
4.34 Struktura zalednega sistema MODx.	59
5.1 Diagram primerov uporabe uporabniškega vmesnika.	62
5.2 Diagram primerov uporabe zalednega vmesnika.	63
5.3 Urejanje sistemskih nastavitev znotraj zalednega vmesnika. . .	64
5.4 Urejanje nastavitev za določen kontekst znotraj zalednega vmesnika	64
5.5 Security nastavitve znotraj zalednega vmesnika.	65
5.6 Upravljanje s pravicami za skupine uporabnikov	65
5.7 Primer urejanja drevesne strukture glavnega menija in podmenijev na uporabniškem vmesniku.	66
5.8 Primer urejanja vsebine poljubne strani v zalednem vmesniku. .	67
5.9 Urejanje kode za administratorje v zalednem vmesniku.	67
5.10 Primer nalaganja CSS datoteke v datotečni sistem MODx. . .	68
5.11 Primer nalaganja slik preko uporabniškega vmesnika.	70
5.12 Uporaba Package Managementa komponente, vgrajene v MODx	70
5.13 Glavni meni zalednega vmesnika za administratorje	71
5.14 Zelo omejen glavni meni oziroma celotni zaledni vmesnik posebne skupine uporabnikov, ki uporabljajo samo Supervizor komponento.	71

5.15	prijava v sistem preko uporabniškega vmesnika	72
5.16	Vstopna stran CEF: vidnih več vsebin (npr. “oranžni dogodki”, ki še niso javni) in ikone za urejanje dinamičnih vsebin, na voljo samo za prijavljene.	72
5.17	Stran dogodka: “hitra” ikona za urejanje (zgoraj levo v admin plošči), na voljo samo za prijavljene.	72
5.18	Urejanje preko uporabniškega vmesnika: preprost vmesnik, a hiter in učinkovit.	73
5.19	Zaledna prijava v sistem.	74
5.20	Prikaz nadzorne plošče (Dashboard) takoj po zaledni prijavi. .	74
5.21	Meni komponent (Components).	74
6.1	Testno okolje: nova različica MODx-a, testiramo tudi nov lasten Plugin, ki doda novo funkcionalnost: Save Draft, Load Draft, Preview Draft (v vmesniku gumbi zgoraj).	76
C.1	Mobile Portrait (320 X 480)	104
C.2	Small Tablet Portrait (600 X 800)	105
C.3	Tablet Portrait (768 X 1024)	106

Tabele

2.1	Glavni elementi spletne strani MODx [20]	7
2.2	Ostali elementi ali značke spletne strani MODx [20].	8
4.1	Seznam preprostih javnih zalednih komponent oziroma dodatka brez grafičnega vmesnika, ki smo jih namestili v naš sistem preko Package Managementa in jih tudi pridno uporabili.	38

Izseki kod

3.1	Primer HTML strukture enostavne spletne strani.	16
3.2	Primer enostavnega CSS.	16
3.3	Primer enostavne PHP skripte, vključene v HTML.	17
3.4	Primer enostavne JavaScript skripte, vključene v HTML. . . .	18
3.5	Primer enostavne strukture SELECT stavka.	18
3.6	Naš primer AJAX kode z uporabo knjižnice jQuery.	19
4.1	Shema za Contacts komponento.	40
4.2	To je zgeneriran contact.map.inc.php, ki je preslikava objekta Contract iz sheme modela v PHP model.	42
4.3	Šolski primer Snippeta, ki vrača kontakte, ločene z vejico, kot niz.	43
4.4	Zelo skrajšana in poenostavljena Ext JS koda ogrodja za pri- kaz kontaktov.	44
4.5	Shema za Web Documents komponento.	48
4.6	Shema za Supervizor komponento.	52
5.1	Konkreten primer uporabe Snippeta combineStyleFiles v kom- binaciji myCache Snippeta znotraj Template.	68
5.2	Konkreten primer uporabe lastnega labPlaceholder Snippeta znotraj vsebine posebne strani.	69
A.1	Predloga (Template) vsake strani spletnega mesta CEF; koda HTML z značkami MODx-a.	95
B.1	Vsebina (Content) strani Staff spletnega mesta CEF; koda HTML z značkami MODx-a	99

Literatura

- [1] (2013) People's Choice Winner for Best Open Source CMS, dostopno na: <http://www.cmscritic.com/2013-people-s-choice-winner-for-best-open-source-cms>
- [2] (2012) Critic's Choice for Best Open Source CMS, dostopno na: <http://www.cmscritic.com/critics-choice-for-best-open-source-cms-goes-to>
- [3] (2014) Linux Ubuntu, dostopno na: <http://sl.wikipedia.org/wiki/Ubuntu>
- [4] (2014) Nginx, dostopno na: <http://en.wikipedia.org/wiki/Nginx>
- [5] (2014) Nginx, dostopno na: <http://wiki.nginx.org/Main>
- [6] (2014) PHP (prevajalnik), dostopno na: <http://sl.wikipedia.org/wiki/PHP>
- [7] (2014) PHP-FPM, dostopno na: <http://php-fpm.org>
- [8] (2014) MySQL, dostopno na: <http://sl.wikipedia.org/wiki/MySQL>
- [9] (2014) MODx Revolution, dostopno na: <http://modx.com>
- [10] (2014) xPDO, dostopno na: <http://xpdo.org>
- [11] (2014) PDO, dostopno na: <http://si1.php.net/pdo>
- [12] (2014) jQuery, dostopno na: <http://jquery.com>

-
- [13] (2014) Ext JS, dostopno na: <http://www.sencha.com/products/extjs>
 - [14] (2014) Foundation, dostopno na: <http://foundation.zurb.com>
 - [15] (2014) HTML, dostopno na: <http://sl.wikipedia.org/wiki/HTML>
 - [16] (2014) CSS, dostopno na: <http://sl.wikipedia.org/wiki/CSS>
 - [17] (2014) JavaScript, dostopno na: <http://sl.wikipedia.org/wiki/JavaScript>
 - [18] (2014) SQL, dostopno na: <http://sl.wikipedia.org/wiki/SQL>
 - [19] (2014) AJAX, dostopno na: [http://sl.wikipedia.org/wiki/Ajax_\(programiranje\)](http://sl.wikipedia.org/wiki/Ajax_(programiranje))
 - [20] (2014) Tag Syntax, dostopno na: <http://rtfm.modx.com/revolution/2.x/making-sites-with-modx/tag-syntax>
 - [21] (2014) Creating an efficient MODx Template, dostopno na: <http://www.virtudraft.com/blog/creating-an-efficient-modx-template.html>
 - [22] (2014) Template, dostopno na: <http://rtfm.modx.com/revolution/2.x/making-sites-with-modx/structuring-your-site/templates>
 - [23] (2014) Resource Field, dostopno na: <http://rtfm.modx.com/revolution/2.x/making-sites-with-modx/structuring-your-site/resources>
 - [24] (2014) Template Variable, dostopno na: <http://rtfm.modx.com/revolution/2.x/making-sites-with-modx/customizing-content/template-variables>
 - [25] (2014) Chunk, dostopno na: <http://rtfm.modx.com/revolution/2.x/making-sites-with-modx/structuring-your-site/chunks>

-
- [26] (2014) Snippet, dostopno na: <http://rtfm.modx.com/revolution/2.x/developing-in-modx/basic-development/snippets>
 - [27] (2014) Plugin, dostopno na: <http://rtfm.modx.com/revolution/2.x/developing-in-modx/basic-development/plugins>
 - [28] (2014) System Settings, dostopno na: <http://rtfm.modx.com/revolution/2.x/administering-your-site/settings/system-settings>
 - [29] (2014) /core/model/schema/modx.mysql.schema.xml, dostopno na: <https://github.com/modxcms/revolution/blob/master/core/model/schema/modx.mysql.schema.xml>
 - [30] (2014) xPDO Schema Visualizer, dostopno na: <http://schemaviewer.dev.kenters.com>
 - [31] (2014) MODx Extras, dostopno na: <http://modx.com/extras>
 - [32] (2014) jCrop, dostopno na: <http://deepliquid.com/content/Jcrop.html>
 - [33] (2014) Fotorama, dostopno na: <http://fotorama.io>
 - [34] (2014) Nginx + PHP-FPM + MySQL + phpMyAdmin on Ubuntu, dostopno na: <http://www.lonelycoder.be/nginx-php-fpm-mysql-phpmyadmin-on-ubuntu-12-04>
 - [35] (2014) MODx Download, dostopno na: <http://modx.com/download>
 - [36] (2014) BannerY, dostopno na: <http://rtfm.modx.com/extras/revo/bannery>
 - [37] (2014) DitsNews, dostopno na: <http://rtfm.modx.com/extras/revo/ditsnews>
 - [38] (2014) FormIt Hook, dostopno na: <http://rtfm.modx.com/extras/revo/formit/formit.hooks>

-
- [39] (2014) FormSave, dostopno na: <http://rtfm.modx.com/extras/revo/formsave>
 - [40] (2014) Google Analytics Dashboard Widget, dostopno na: <https://github.com/Sterc/Analytics-dashboard-widget>
 - [41] (2014) Analytics, dostopno na: <http://rtfm.modx.com/extras/revo/analytics>
 - [42] (2014) Janitor, dostopno na: <http://modx.com/extras/package/janitor>
 - [43] (2014) MoreGallery, dostopno na: <https://www.modmore.com/extras/moregallery>
 - [44] (2014) CodeMirror, dostopno na: <http://modx.com/extras/package/codemirror>
 - [45] (2014) FormIt, dostopno na: <http://rtfm.modx.com/extras/revo/formit>
 - [46] (2014) getCache, dostopno na: <http://modx.com/extras/package/getcache>
 - [47] (2014) getCollection, dostopno na: <http://rtfm.modx.com/xpdo/2.x/class-reference/xpdo/xpdo.getcollection>
 - [48] (2014) getPage, dostopno na: <http://rtfm.modx.com/extras/revo/getpage>
 - [49] (2014) getResourceField, dostopno na: <http://rtfm.modx.com/extras/revo/getresourcefield>
 - [50] (2014) getResources, dostopno na: <http://rtfm.modx.com/extras/revo/getresources>
 - [51] (2014) iCalX, dostopno na: <http://modx.com/extras/package/icalx>

-
- [52] (2014) Login, dostopno na: <http://rtfm.modx.com/extras/revo/login>
 - [53] (2014) MultiUploadDialog, dostopno na: <https://github.com/thomasd/MultiUploadDialog>
 - [54] (2014) NewsPublisher, dostopno na: <http://bobsguides.com/newspublisher-tutorial.html>
 - [55] (2014) Output Filters, dostopno na: [http://rtfm.modx.com/revolution/2.x/making-sites-with-modx/customizing-content/input-and-output-filters-\(output-modifiers\)](http://rtfm.modx.com/revolution/2.x/making-sites-with-modx/customizing-content/input-and-output-filters-(output-modifiers))
 - [56] (2014) phpThumbOf, dostopno na: <http://rtfm.modx.com/extras/revo/phpthumbof>
 - [57] (2014) Rowboat, dostopno na: <http://rtfm.modx.com/extras/revo/rowboat>
 - [58] (2014) SimpleSearch, dostopno na: <http://rtfm.modx.com/extras/revo/simplesearch>
 - [59] (2014) translit, dostopno na: <http://modx.com/extras/package/translit>
 - [60] (2014) Image+, dostopno na: <http://modx.com/extras/package/imageplustvinput>
 - [61] (2014) Wayfinder, dostopno na: <http://rtfm.modx.com/extras/revo/wayfinder>
 - [62] (2014) MODx Documentation, dostopno na: <http://rtfm.modx.com>
 - [63] (2014) MODx Forum, dostopno na: <http://forums.modx.com>
 - [64] (2014) freenode IRC, kanal modx, dostopno na: <http://webchat.freenode.net>

-
- [65] (2014) iskalnik Google, dostopno na: <https://www.google.com>
- [66] (2014) Developing an Extra in MODX Revolution, dostopno na: <http://rtfm.modx.com/revolution/2.x/case-studies-and-tutorials/developing-an-extra-in-modx-revolution>
- [67] (2014) Custom Manager Pages, dostopno na: <http://rtfm.modx.com/revolution/2.x/developing-in-modx/advanced-development/custom-manager-pages>
- [68] (2014) Creating a Model With xPDO, dostopno na: <http://rtfm.modx.com/xpdo/2.x/getting-started/creating-a-model-with-xpdo>
- [69] (2014) Reverse Engineer xPDO Classes from Existing Database Table, dostopno na: <http://rtfm.modx.com/revolution/2.x/case-studies-and-tutorials/reverse-engineer-xpdo-classes-from-existing-database-table>
- [70] (2014) Defining Relationships, dostopno na: <http://rtfm.modx.com/xpdo/2.x/getting-started/creating-a-model-with-xpdo/defining-a-schema/defining-relationships>
- [71] (2014) EMRIS, PREDSTAVITEV STRUKTURNEGA RAZVOJA IS, Razvojni modeli, dostopno na: <http://www2.gov.si/mju/emris.nsf/0/E1FA1E8B375BF310C1256E9E003EA512?OpenDocument>
- [72] (2014) EMRIS, PREDSTAVITEV STRUKTURNEGA RAZVOJA IS, Razvojni modeli, Inkrementalni model, dostopno na: <http://www2.gov.si/mju/emris.nsf/d4b65258c7524acfc1256e9e003c292a/89c0cd312c89ec71c1256e9e003eb479?OpenDocument>
- [73] (2014) Creating a 3rd Party Component Build Script, dostopno na: <http://rtfm.modx.com/revolution/2.x/developing-in-modx/advanced-development/package-management/creating-a-3rd-party-component-build-script>

-
- [74] (2014) Transport Package, dostopno na: <http://rtfm.modx.com/revolution/2.x/developing-in-modx/advanced-development/package-management/transport-packages>
- [75] (2014) Getting to know Class-based Processors in MODX 2.2, dostopno na: <https://www.markhamstra.com/xpdo/2012/getting-started-with-class-based-processors-2.2>
- [76] (2014) Using runProcessor, dostopno na: <http://rtfm.modx.com/revolution/2.x/developing-in-modx/advanced-development/using-runprocessor>
- [77] (2011) Bob Ray. MODX: The Official Guide. MODX Press
- [78] (2000) Marin Slilič, Marjan Krisper, Ivan Rozman. EMRIS - Enotna metodologija razvoja informacijskih sistemov. Zv. 4, Objektni razvoj. Ljubljana : Vlada Republike Slovenije, Center vlade RS za informatiko, 88 str.

Dodatek A

Predloga (Template) strani spletnega mesta CEF

```
1 <!DOCTYPE html>
2 <!--[if lt IE 7]> <html class="no-js lt-ie9 lt-ie8 lt-ie7"
   lang="[[++cultureKey]]"><h4 class="red">This Browser is not supported!
   Please upgrade (min IE8)</h4><![endif]-->
3 <!--[if IE 7]> <html class="no-js lt-ie9 lt-ie8"
   lang="[[++cultureKey]]"><![endif]-->
4 <!--[if IE 8]> <html class="no-js lt-ie9" lang="[[++cultureKey]]">
   <![endif]-->
5 <!--[if gt IE 8]><!--> <html class="no-js" lang="[[++cultureKey]]">
   <!--<![endif]-->
6
7 <head>
8   <base href="[[!++site_url]]" />
9   <meta charset="[[++modx_charset:lowercase]]" />
10  <meta name="author" content="[[*createdby:userinfo='fullname']]" />
11  <meta name="keywords" content="" />
12  <meta http-equiv="X-UA-Compatible" content="IE=edge" />
13  <meta name="viewport" content="width=device-width,initial-scale=1">
14
15  <title>[[*pagetitle]] - [[++site_name]]</title>
16
17  <link rel="shortcut icon" href="[[++base_url]]favicon.ico"
    type="image/x-icon" />
18
19  [[!myCache?
20    &snippet='combineStyleFiles'
```

```

21     &myCacheFolder='mycss '
22     &key='myCSS '
23
24     &targetPath='assets/site/'
25     &targetFilename='screen.v5.css '
26
27     &files='assets/templates/foundation4/css_new/normalize.min.css,
28             assets/templates/foundation4/css_new/foundation.min.css,
29             assets/site/simple.css,
30             assets/site/css/styles.less,
31             assets/js/meanmenu/meanmenu.css '
32 ]]
33
34 <!--[if lt IE 9]><link rel="stylesheet"
35     href="[[++base_url]]assets/site/ie.css" /><![endif]-->
36
37 <!--[if lt IE 9]>
38 <script src="[[++base_url]]assets/js/html5.js"></script>
39 <script src="[[++base_url]]assets/js/respond.min.js"></script>
40 <![endif]-->
41
42 <!-- RSS -->
43 <link rel="alternate" type="application/rss+xml" title="Follow this
44     website with RSS" href="[[~363]]" />
45 <!-- GA -->
46 [[!Analytics? &setAccount='UA-XXXXXXX-1 '
47     &excludeLoggedInUserContextList='mgr,web,web.si']]
48 </head>
49
50 <body style="">
51
52 <!-- sticky footer mod -->
53 <div id="wrapper">
54
55 <!-- background -->
56 <div id="bg_left"></div>
57 <div class="row cefbg" style="min-height:1200px;">
58 <div class="large-9 columns">
59
60 <noscript>
61 <div class="noscript rounded">
62     ENABLE JAVASCRIPT in your browser for optimal viewing.
63 </div>
64 </noscript>
65
66 [[!$header]]

```

```

65     <div class="container">
66
67         <div class="row">
68             <div class="large-9 columns" id="menMenuTarget">
69                 </div>
70         </div>
71
72         <div class="row">
73             <div class="large-9 columns">
74                 [[BreadCrbm]]
75             </div>
76         </div>
77
78
79         <div class="row">
80             <div class="large-9 columns">
81                 [[!contentTemplate]]
82             </div>
83         </div>
84
85     </div> <!-- end of container -->
86
87 </div>
88 </div>
89
90 <div class="push"></div>
91
92
93 </div> <!-- end of sticky wrap -->
94
95 [[!$footer]]
96
97
98 <div id="adminPanel">
99     [[!LoginNew]] <!-- must be uncached -->
100 </div>
101
102 <script type="text/javascript"
103     src="[[++base_url]]assets/js/LAB.min.js"></script>
104
105 [[!labPlaceholder? <!-- must be uncached -->
106     $js='assets/js/blazy.min.js,
107         assets/js/meanmenu/jquery.meanmenu.better.js,
108         assets/js/jquery.cookiecuttr.cookie.v5.min.js'
109
110     $inline='[[!$activeLeaf]]; var bLazy;'
```

```
111     &chunk='menuHeaderForTouch ,  
112           stickyZurbFooter ,  
113           loadMeanMenu ,  
114           jqPlaceholder ,  
115           cookie ,  
116           fn.loadcssfile ,  
117           loadPrintCss '  
118 ]]  
119  
120 [[!addSpecialJS-CSS]]  
121 [[!do.stuff.if.admin]]  
122 [[!labJS-CSS]]  
123  
124 </body>  
125 </html>
```

Izsek kode A.1: Predloga (Template) vsake strani spletnega mesta CEF; koda HTML z značkami MODx-a.

Dodatek B

Vsebina (Content) strani Staff spletnega mesta CEF

```
1
2 <div class="row">
3   <div class="large-9 columns">
4     <div class="pageHeader" style="margin-bottom:20px;">[[*pagetitle]]</div>
5   </div>
6 </div>
7
8 <div class="row">
9
10   <div class="large-2 columns">
11
12     <!-- navigation -->
13     <div style="" class="uniTabsNav" id="filtersNav">
14
15       <div class="tabs firstTab level1 activeTab" title="">
16         <a class="filterSelected" data-filter="*">Show all staff</a>
17       </div>
18
19       <div class="tabs level1" title="">
20         <a data-filter=".ma">Management and administration</a>
21       </div>
22
23       <div class="tabs level1" title="">
24         <a data-filter=".it">IT support</a>
25       </div>
26
```

```

27     <div class="tabs level1" title="">
28         <a data-filter=".pap">Programs and projects</a>
29     </div>
30
31     <div class="tabs level1" title="">
32         <a data-filter=".imf">IMF technical assistance</a>
33     </div>
34
35 </div>
36
37 </div>
38
39 <div class="large-7 columns">
40
41 <!-- content -->
42 <div class="" id="contacts"> <!-- isotope class removed -->
43 <ul class="small-block-grid-2 medium-block-grid-3 large-block-grid-4">
44
45     [[myCache?
46         &snippet='Rowboat '
47         &key='myContacts '
48
49         &table='modx_contacts '
50         &tpl='contact_tpl '
51         &limit= '0'
52         &where='{ "visible": "1" }'
53         &cacheResults='0'
54         &sortBy='order '
55     ]]
56
57 </ul>
58 </div>
59
60 </div>
61
62
63 </div>
64
65
66 [[!labPlaceholder?
67     &js='assets/js/jquery.slimscroll.min.js'
68     &chunk='bLazy,slimscrollContacts '
69 ]]
70
71 [[!labPlaceholderChunk?
72     &chunk='flipCard '
73     &selector='.card '

```

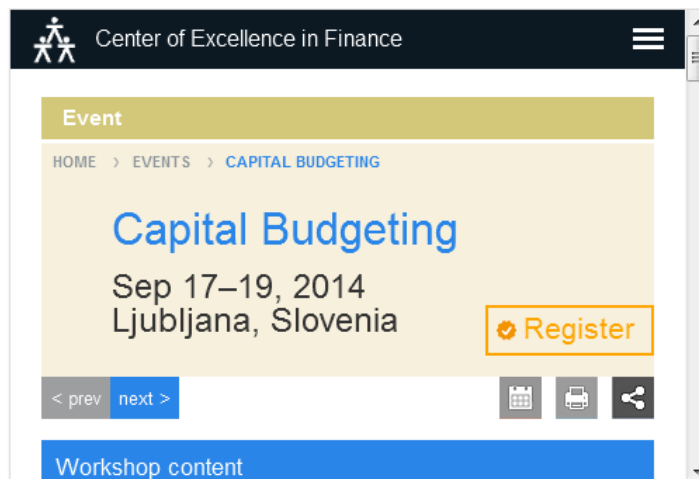


```
74     &container='contacts'
75 ]]
76
77
78 [[!labPlaceholderChunk?
79     &chunk='disableScrollOnHover'
80     &selector='.back'
81 ]]
```

Izsek kode B.1: Vsebina (Content) strani Staff spletnega mesta CEF; koda HTML z značkami MODx-a

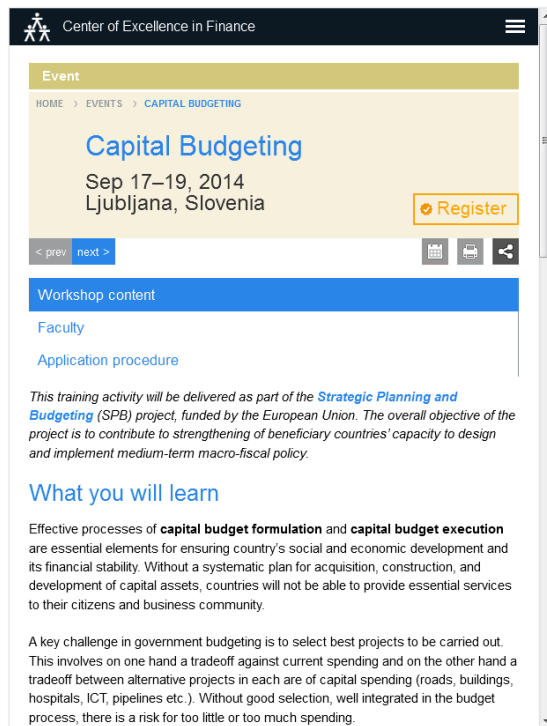
Dodatek C

Prikaz strani nekega dogodka
spletnega mesta CEF za
naprave z različnimi
ločljivostmi zaslona

▼ Mobile portrait (320x480)**▼ Mobile landscape (480x320)**

Slika C.1: Mobile Portrait (320 X 480)

▼ Small tablet portrait (600x800)

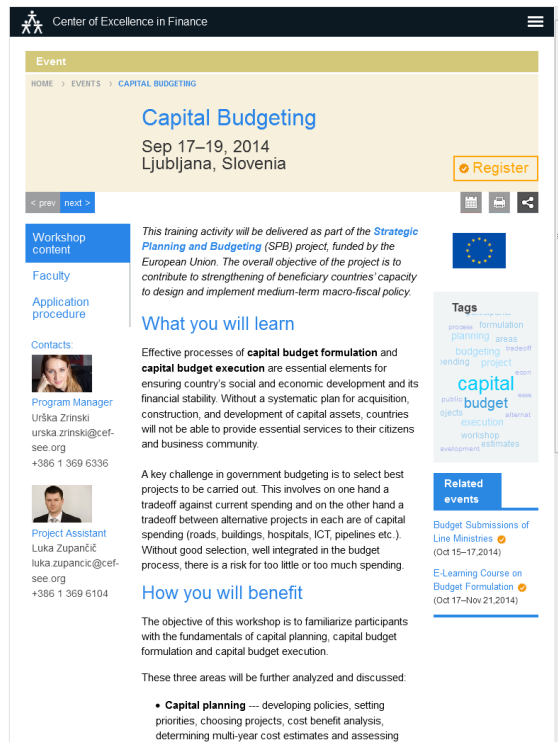


▼ Small tablet landscape (800x600)

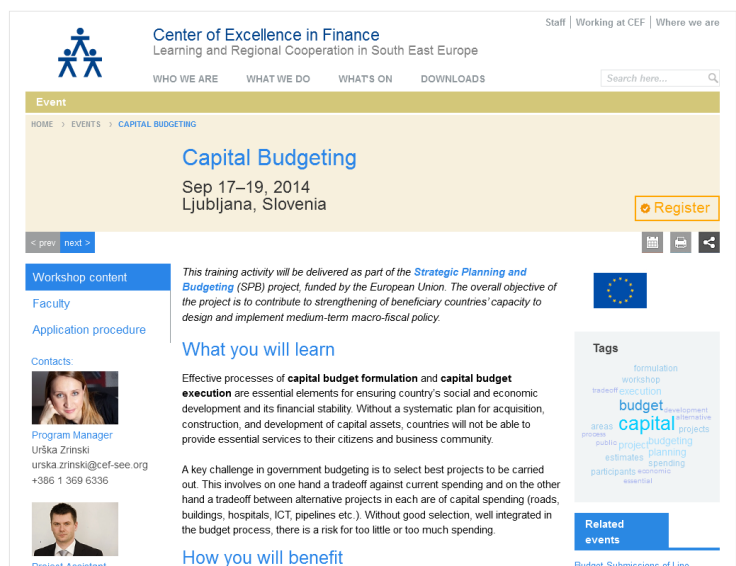


Slika C.2: Small Tablet Portrait (600 X 800)

▼ Tablet portrait (768x1024)



▼ Tablet landscape (1024x768)



Slika C.3: Tablet Portrait (768 X 1024)